COMPUTING FOR HOME AND BUSINESS

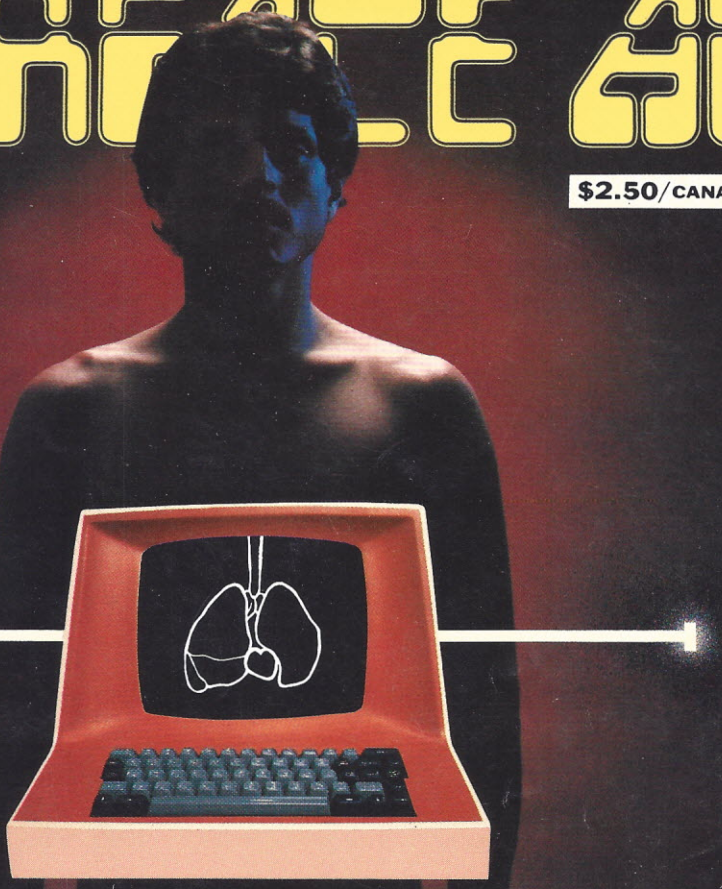
SEPTEMBER 1980

INTERFACE AGE™

$2.50/CANADA/MEXICO $3.00

IDs LOOK TO MICROS

Lung Action: Charting the Deep Breath



Nerve Waves: A Way to Plug In

SELECTING SOFTWARE

Do's and Don'ts

Custom Programs "Beat the System"

Hard and Fast...

...Bulk Storage from Industrial Micro Systems

THE NEW MODEL 16

The new Industrial Micro Systems Model 16 Hard Disk Subsystem is a "fixed-removable" high speed, bulk storage device providing from 32 megabytes (32 million characters) to 96 megabytes of on-line storage for the Industrial Micro Systems 8000 or Series 5000 microcomputer systems. The Model 16 includes a credenza enclosure that provides a quiet, strong and attractive package for office or industrial applications where large memory is required. The Model 16 also includes a fully buffered DMA S-100 bus controller for fast and easy interfacing.

WINCHESTER TECHNOLOGY WITH BUILT-IN BACKUP

The Model 16 includes a 16 megabyte removable cartridge and a 16, 48, or 80 megabyte fixed media that employs Winchester 3340 technology. Files and programs may be copied between the fixed media and the removable cartridge for fast, easy backup and archival storage.

FAST ACCESS

The interface between the Model 16 hard disk and the Industrial Micro Systems computer is provided by the Hard Disk Controller. The Hard Disk Controller utilizes Direct Memory Access (DMA) for fast data transfer with minimum processor intervention. The maximum data transfer rate is 1.2 megabytes per second and the controller fully buffers the data, a sector at a time, to and from the disk. **Available in 220 V, 50 HZ Versions**

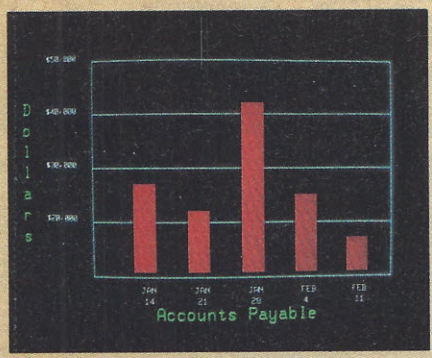
Now you don't have to look hard for fast computing power. Contact your Industrial Micro Systems Dealer today.

INDUSTRIAL MICRO SYSTEMS

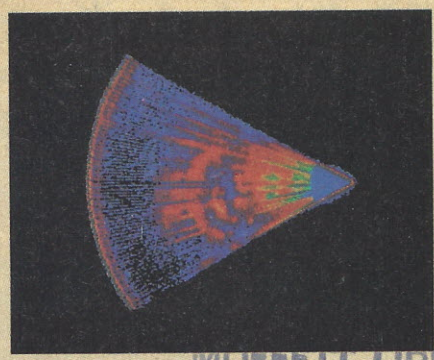
Marketing
628 N. Eckhoff, Orange, CA 92668
(714) 978-6966

Manufacturing
2800 Lockheed Way, Carson City, NV 89701
(702) 883-7611

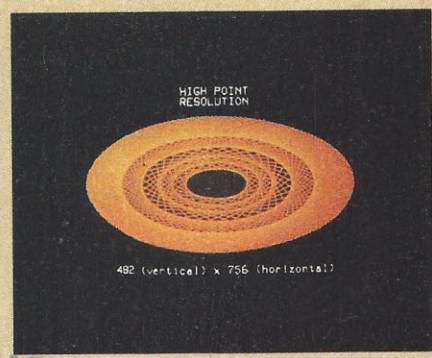
CIRCLE INQUIRY NO. 28



Management Information Display



Ultrasonic heart sector scan



High-resolution display with alphanumerics

Get the professional color display that has BASIC/FORTRAN simplicity

LOW-PRICED, TOO

Here's a color display that has everything: professional-level resolution, enormous color range, easy software, NTSC conformance, and low price.

Basically, this new Cromemco Model SDI* is a two-board interface that plugs into any Cromemco computer.

The SDI then maps computer display memory content onto a convenient color monitor to give high-quality, high-resolution displays (756 H x 482 V pixels).

When we say the SDI results in a high-quality professional display, we mean **you can't get higher resolution than this system offers in an** NTSC-conforming display.

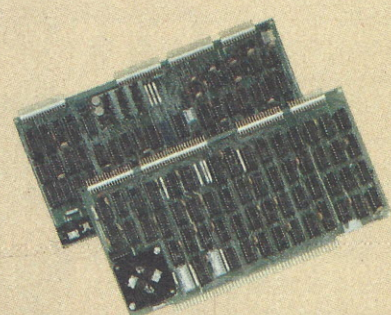
The resolution surpasses that of a color TV picture.

BASIC/FORTRAN programming

Besides its high resolution and low price, the new SDI lets you control with optional Cromemco software packages that use simple BASIC- and FORTRAN-like commands.

Pick any of 16 colors (from a 4096-color palette) with instructions like DEFCLR (c, R, G, B). Or obtain a circle of specified size, location, and color with XCIRC (x, y, r, c).

*U.S. Pat. No. 4121283



Model SDI High-Resolution Color Graphics Interface

HIGH RESOLUTION

The SDI's high resolution gives a professional-quality display that strictly meets NTSC requirements. You get 756 pixels on every visible line of the NTSC standard display of 482 image lines. Vertical line spacing is 1 pixel.

To achieve the high-quality display, a separate output signal is produced for each of the three component colors (red, green, blue). This yields a sharper image than is possible using an NTSC-composite video signal and color TV set. Full image quality is readily realized with our high-quality RGB Monitor or any conventional red/green/blue monitor common in TV work.

Model SDI plugs into Z-2H 11-megabyte hard disk computer or any Cromemco computer

DISPLAY MEMORY

Along with the SDI we also offer an optional fast and novel **two-port** memory that gives independent high-speed access to the computer memory. The two-port memory stores one full display, permitting fast computer operation even during display.

CONTACT YOUR REP NOW

The Model SDI has been used in scientific work, engineering, business, TV, color graphics, and other areas. It's a good example of how Cromemco keeps computers in the field up to date, since it turns any Cromemco computer into an up-to-date color display computer.

The SDI has still more features that you should be informed about. So contact your Cromemco representative now and see all that the SDI will do for you.

Cromemco
incorporated
280 BERNARDO AVE., MOUNTAIN VIEW, CA 94040 • (415) 964-7400
Tomorrow's computers today

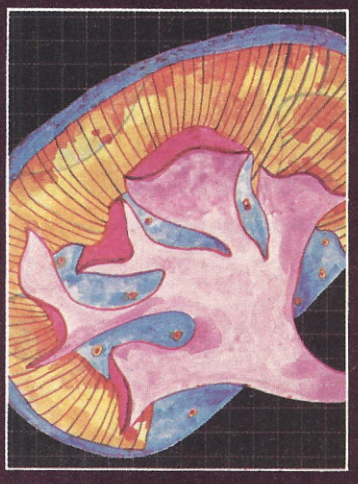
CIRCLE INQUIRY NO. 12

VOL. 5, ISSUE 9
SINCE DECEMBER 1975

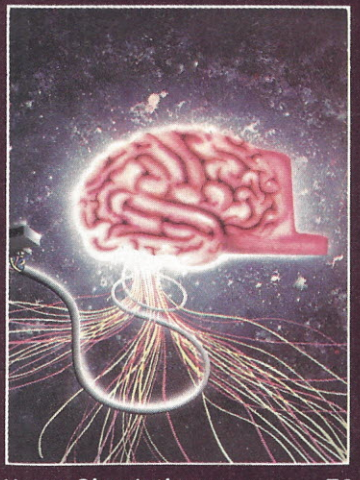
SEPTEMBER 1980

INTERFACE AGE™

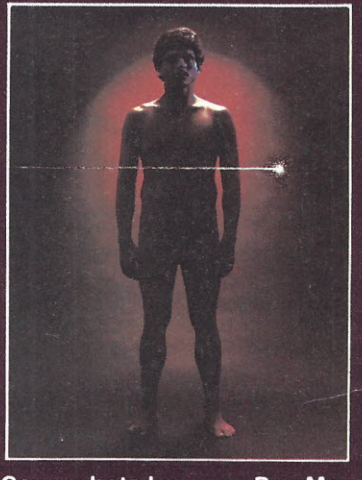
COMPUTING FOR HOME AND BUSINESS APPLICATIONS



Spirometric Calculation . . . 14



Nerve Simulation 70



Cover photo by . . . . . Don May
Model . . . . . Richard Ortiz

FEATURES

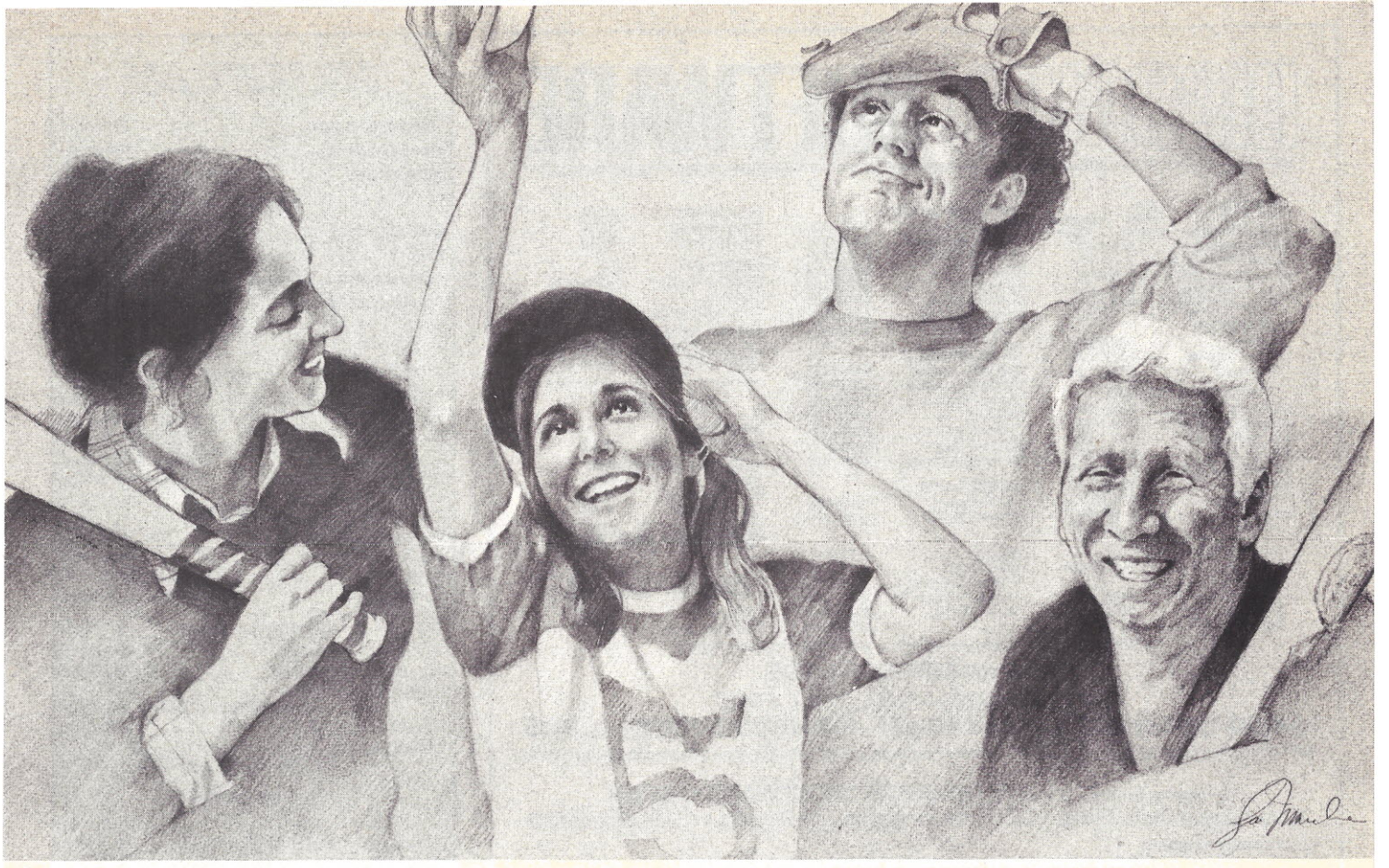
COLUMNS

DEPARTMENTS

Contact authors of monthly columns by writing them at INTERFACE AGE, P.O. Box 1234, Cerritos, CA 90701 in care of their respective columns.

INTERFACE AGE Magazine, published monthly by McPheters, Wolfe & Jones, 16704 Marquardt Ave., Cerritos, CA 90701. Subscription rates: U.S. $18.00, Canada/Mexico $20.00, all other countries $28.00. Make checks payable in U.S. funds drawn on a U.S. bank. Opinions expressed in by-lined articles do not necessarily reflect the opinion of this magazine or the publisher. Mention of products by trade name in editorial material or advertisements contained herein in no way constitutes endorsement of the product or products by this magazine or the publisher. Circulation Department, (213) 926-9540.

INTERFACE AGE Magazine is catalogued in the Library of Congress, Classification No. QA75.5.155. USPS No. 528150. ISSN Publication No. 0147-2992. Membership in Audit Bureau of Circulations applied for.
POSTMASTER: Please send change of address form 3579 and undelivered copies to **INTERFACE AGE** Magazine, 16704 Marquardt Ave., Cerritos, CA 90701. Controlled circulation postage paid at Olive Branch, Mississippi and Artesia, California.



Here's to your health! Six Vitafacts™ programs for you and your family.

Being healthy and happy is so very important. And now you can use your Apple,™ PET/CBM™ or TRS-80™ computer to help stay that way.

Introducing six new Personal Software™ Vitafacts Series programs for your health: Growing Up, Heart Attacks, Talking About Sex, Drinking & Drugs, Birth Control, and Your Blood Pressure.

Using a Vitafacts program is like getting advice from your family doctor. Accurate, up-to-date medical information about physical and mental health, presented in a friendly, straightforward way. You'll feel even more confident knowing that each program is approved and endorsed by The College of Family Physicians in Canada, where the Vitafacts Series is created by The Richmond Software Group and Medifacts Ltd.

Each program includes a manual with diagrams and glossary, a computer cassette, and an audio cassette. The audio tape uses short dramas and straight talk to tell the story. Then the computer exercises verify your new knowledge. And because you use your knowledge right away, you remember more. It's fun! And a great way to learn.

Growing Up helps families cope with the teenage years. Teenagers learn more about their physical and emotional changes, and parents learn more about helping their children through it all.

Heart Attacks describes what one is, how to recognize it, what to do when someone is experiencing one, and—most important—how to help prevent one.

Talking About Sex, presented by two of Canada's foremost sex counselors, offers to adults the proper information and appropriate attitude for a satisfying sex life.

Drinking & Drugs. No lecturing or talking down. Just straight facts about the very real dangers of alcohol and drugs. Prepared for teenagers, but good for adults.

Birth Control explains clearly and completely conception, birth and prevention of unwanted pregnancies. Important information for teenagers, and helpful for adults.

Your Blood Pressure. No one is immune to the risks of high blood pressure. Knowledge is your best defense, and this program has the information.

"Vita" means "Life" in Latin. We hope these programs make your life healthier and happier.

Retail price is just $19.95.

Ask your Personal Software dealer for a demonstration, and for our new catalog. Call or write to find your nearest dealer. (408) 745-7841. Personal Software Inc., 1330 Bordeaux Dr., Sunnyvale, CA 94086.

PERSONAL SOFTWARE

CIRCLE INQUIRY NO. 52

VITAFACTS SERIES

TM—Vitafacts is a trademark of Medifacts Ltd.; Apple is a trademark of Apple Computer, Inc.; PET is a trademark of Commodore Business Machines, Inc.; TRS-80 is a trademark of the Radio Shack Div.

INTERFACE AGE

Robert S. Jones — Publisher
Nancy A. Jones — Executive Publisher
Mike Antich — Publication Director

Editorial

Managing Editor — **Frank Vaughan**
Editor — **Michael Panchak**
Assistant Editors — **Kathy Tekawa, Les Spindle**
Contributing Editors — **Alfred Adler, Bob Albrecht, Al Baker, R. W. Bemer, Roger Edelson, Tom Fox, Roger C. Garrett, Carl Heintz, Elliott MacLennan, Alan R. Miller**

Production

Production Manager — **Terri Ledesma**
Art Director — **Fino Ortiz**
Artist — **Samantha Lee**
Typographer — **Melody A. Martens**

Administration

Circulation Manager — **Colin Cato**
Accounting Supervisor — **Kay Soto**
Publication Assistants — **Doris Riopel, Cheryl Johnston**
Circulation Assistant — **Lilly Lisa**
Accounting Assistants — **Sharon Spolar, Mary Ann Lower**

Advertising

New England Region — **Dick Green**
7 Lincoln St., Wakefield, MA 01880
(617) 245-9105

Eastern Region — **John Sensenstein**
20 Community Pl., Ste. 140, Morristown, NJ 07960
(201) 267-3032

Southeast Region — **Harry Dill**
3938 Sussex Avenue, Charlotte, NC 28210
(704) 552-1004

Midwest Region — **Al Gravenhorst, Steve Skinner**
5901 N. Cicero Ave., Chicago, IL 60646
(312) 545-8621

Western Region — **Mike Antich**
P.O. Box 1234, Cerritos, CA 90701
(213) 926-9544

Japan — **Tomoyuki Inatsuki**
Trade Media Japan Inc., R. 212 Azabu Hts., 1-5-10, Roppongi, Minato-ku, Tokyo 106
Telephone: (03) 585-0581 Telex: J28208

Interface Age Europe

Director, European Operations — **H.L. Grohmann**
Dahlienstr. 4, D-8011 Munchen-Vaterstetten
West Germany
Telephone: 08106/7396

International Newsstand Distribution

Director — **Lew Ullian**
Orberstrasse 38, D-6000 Frankfurt/M. 61
West Germany
Telephone: (0611) 44 77 90/41 84 80

International/Domestic Retail Circulation

(213) 926-9544 — **Mary Ann Lower**

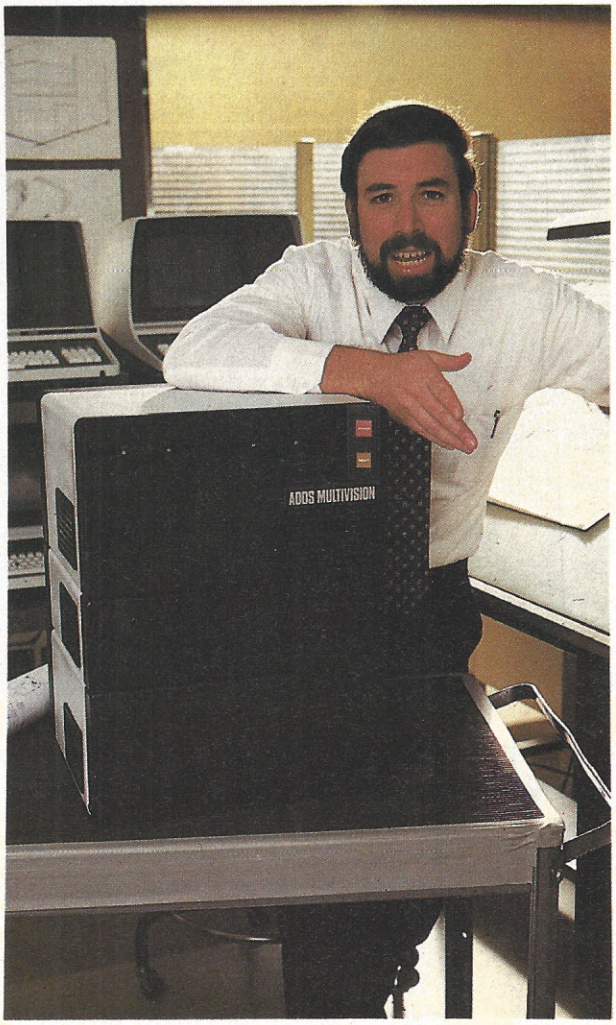
MEMBER OF THE WESTERN PUBLICATIONS ASSOCIATION — wpa

AMERICAN SOCIETY OF BUSINESS PRESS EDITORS — MEMBER

16704 Marquardt, Cerritos, CA 90701
(213) 926-9544
TWX (910) 583-1412

"Now you can multiply your computing power as you grow."

Geof Karlin
Director of Systems Development



"At ADDS, we've just designed a unique computer family that can take you all the way from small to big.

"We call it ADDS Multivision™–a trio of stacking, CP/M®-compatible computers that lets you multiply computing power as needed. Without a change in operating system or programming languages. Without costly conversion. It works like this:

"MULTIVISION 1 (top module) is a get-started computer with 5 MHz processor, 64K bytes of RAM and mini disk storage capacity of 700K bytes. It lists for $3,785 without terminal.

"MULTIVISION 2 (top and bottom modules) provides 5M or 10M bytes of additional hard-disk storage. Priced thousands less than other hard-disk systems, it lists for $7,995 with 5M bytes of disk.

"MULTIVISION 3 (entire stack) tops off the line, giving you a multi-user system with up to 256K bytes of RAM that supports up to four display terminals simultaneously.

"We even offer an ADDS-developed package that lets you use Multivision as a word processor.

"Before you decide upon any small computer, look into ADDS Multivision. For years we've been the largest supplier of display terminals to computer giants. Now we're making a system for you."

For information write: Systems Division, Applied Digital Data Systems Inc., 100 Marcus Boulevard, Hauppauge, N.Y. 11787. Dealer inquiries invited.

CP/M is a registered trademark of Digital Research, Inc.

ADDS

SOMETHING EXTRA IN EVERYTHING WE DO

MULTIVISION

ATARI: PERSONAL COMPUTER

START WITH A BETTER COMPUTER.

Atari personal computers have built-in capabilities you can't even add on to many other personal computers. Three programming formats (slip-in ROM cartridges, disk and cassette). A 57-key upper/lower case ASCII keyboard with 29 keystroke graphics 128 colors symbols.

and hues. Four separate sound channels and a built-in speaker. Four game and graphics controller ports. A built-in RF modulator. FCC approval for connection to any TV. Plus, nationwide Atari Authorized Service Centers. And more.

ADD MEMORY.

The ATARI 800™ comes with 16K of memory. Expand up to a full 48K of memory with 8K or 16K RAM Memory Modules™ you install yourself. In less than a minute. The ATARI 400™ is supplied with 8K of RAM which can be expanded to 16K at Authorized Service Centers. Both models may be expanded to 26K of ROM with slip-in ROM cartridge programs.

ADD PERIPHERALS.

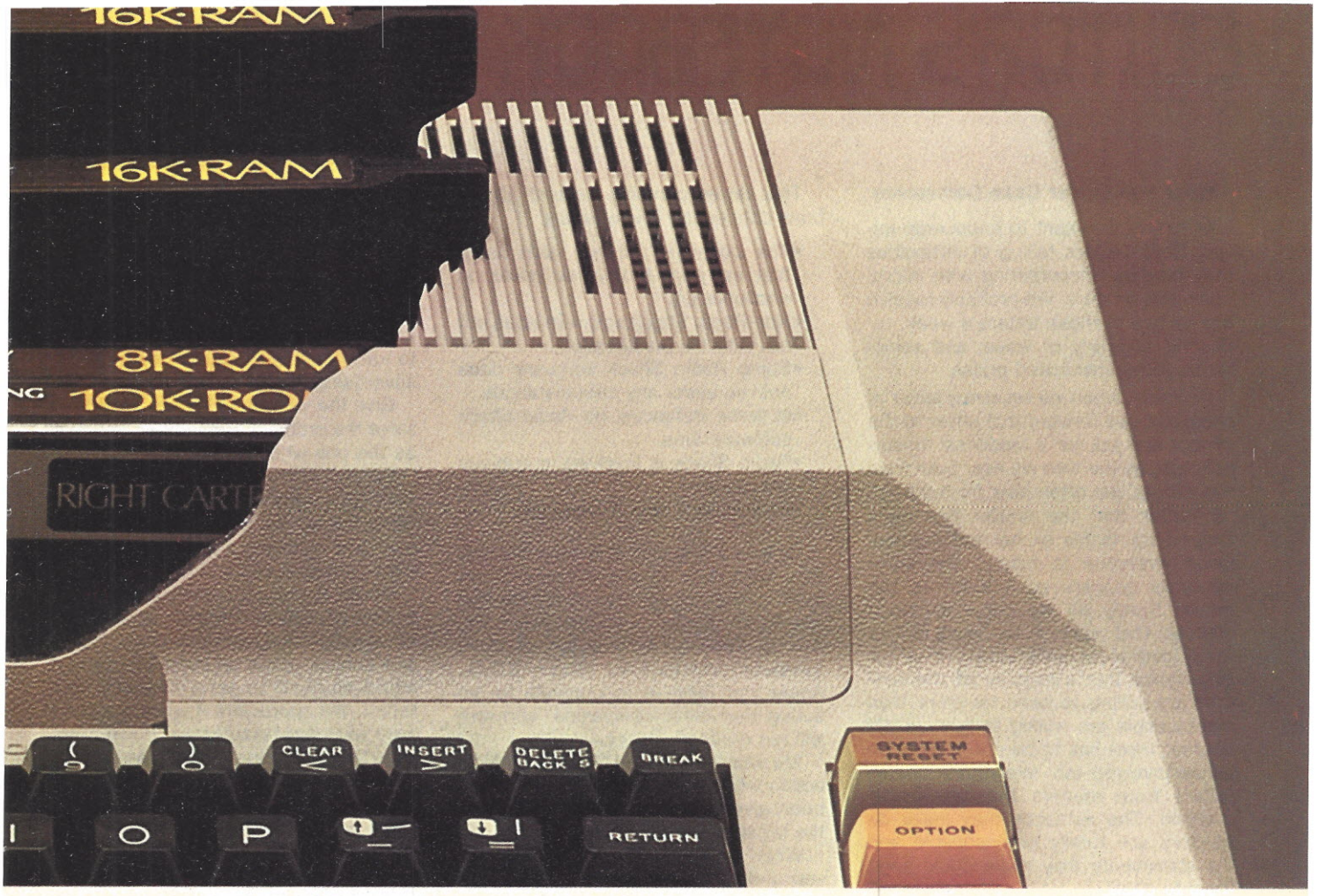
Atari peripherals are specifically designed to complement your system. Add the ATARI 410™ Audio-digital Program Recorder. Add disk drives (single or dual double-density*). The ATARI 800 individually accesses up to four drives. Add the ATARI 850™ RS232 Interface Module. Add high-speed 40 or 80-column dot-matrix printers, or, a 40-column thermal printer. Add an acoustic modem for remote data access. Add a light pen.* And there are more Atari peripherals on the way.

ADD PROGRAMS.

Atari's rapidly expanding software library includes some of the most useful, enlightening and entertaining programs available. (Some programs may require more than 16K RAM.)

- **Personal Finance and Record Keeping.** Sophisticated investment analysis programs, including bond analysis, stock analysis, stock charting and mortgage and loan analysis.

*Available Fall, 1980



SYSTEMS THAT GROW WITH YOU

- **Personal Interest and Development.** Conversational languages.* Music composition. Touch typing. Biorhythm. And more.
- **Professional Applications.** Graphics, statistics and programmable calculator programs. Plus VisiCalc,† the "what if" problem solving program.
- **Education.** ATARI Talk & Teach™ programs offer instruction from primary through college level. Plus, instructive games for all ages.
- **Information & Communication.** Turn your Atari into an interactive terminal. Access stock quotations, news services and other useful data banks. Transmit and receive data over regular phone lines. And more.
- **Entertainment.** Colorful, exciting programs include Star Raiders,™ Chess, Blackjack...and many more.

†Trademark: Personal Software, Inc.

- **Programming Languages.** An ATARI BASIC cartridge is included. An Assembler/ Editor Cartridge is available for the advanced programmer. PILOT, extended BASIC and Pascal will be available in 1981.
- **Small Business Accounting.** General Ledger. Accounts Receivable. Accounts Payable. Inventory Control. Order Entry.

ADD IT UP FOR YOURSELF.

Compare the Atari to any personal computer in its price range. Compare built-in features. Compare expandability. Compare peripherals and accessories. Compare programming and formats (ROM cartridge, disk or cassette). And if you run a business, ask your Atari retailer about the Atari Accountant,™ the complete, computerized accounting system for small businesses. At Atari we're building computer systems you won't outgrow.

ATARI®

PERSONAL COMPUTERS

CIRCLE INQUIRY NO. 3

1265 Borregas Avenue, Sunnyvale, CA 94086
Call toll-free (800) 538-8547 (Except Alaska and Hawaii)
(In California: (800) 672-1404) for the name of your nearest Atari retailer.

EDITOR'S NOTEBOOK

Upper and Lower Case Comments

IA makes it a point to encourage letters from readers telling of difficulties they may be encountering with microcomputer vendors. We probably receive one or two of these letters a week, reflecting a variety of woes, and sometimes giving unsolicited praise.

Normally, when we receive a letter of complaint, we forward that letter to the vendor and ask for a response. Unsurprisingly, by the time we hear back from the vendor, we often hear from the letter writer that the matter has been cleared up to his or her satisfaction. Once everyone is happy, the letter generally appears as a letter to the editor. Rarely do we come across a dispute that cannot be resolved to some degree of satisfaction. It is our experience that a majority of the vendors are willing to bend far more than most people are willing to believe. Of course, there has to be some common sense involved too. We often receive letters from readers that are hardly credible. The demands they make on vendors are totally unreasonable and the statements they make could send them to court.

Following is an actual letter we received:

"I would like to call your attention to a very serious matter that Radio Shack has created.

"Many of the computer magazines are carrying full page ads from Radio Shack for a lower case modification for the TRS-80. It costs about $99. I had it done and got my TRS-80 back in about a week. However, when I looked at the instruction book I found a very serious omission by Radio Shack.

"After you have had this modification done, you no longer can run any non-Radio Shack software. You can't even run some Radio Shack software. If you load the software to run the lower case driver, you can't even run any Radio Shack software.

"Radio Shack didn't even think it of sufficient concern to put in their ad or to even warn you when you order the modification. I took my keyboard back and lost money doing it. I got my money refunded but had put decals from Tshort on my keyboard. I now have to purchase new decals.

"I think your readers should be warned about this very dubious business practice. Also, I think people who purchased the modification and had to return their keyboard [should] band together to file a class action suit against Radio Shack."

This letter contains several allegations concerning Radio Shack:

- The company did not warn users that software difficulties could be encountered.
- Non-Radio Shack software does not run with the modification.
- Some Radio Shack software does not run under any circumstances.
- In some instances, no Radio Shack software runs.
- Radio Shack is involved in dubious practices.
- Radio Shack should be sued.

When we first look at a letter such as this, we check to see if there is any logic to the gripes. The complaint that non-Radio Shack software won't run can be dismissed immediately since no computer vendor—not Radio Shack, Apple, nor IBM—is responsible for insuring that other companies' software will run on its machines.

We also tend to dismiss calls for legal action when the individual involved has been given a refund by the vendor, as the letter-writer admits.

When we contacted Radio Shack, it responded to several of the allegations with the following:

- Several pieces of nationally distributed advertising that contain a notice of possible software limitations. It also pointed out that the advertisement mentioned by the complaintant was an ad for Scriptsit software, and referred to five or six different pieces of hardware.
- Radio Shack alleges that non-Radio Shack software will run with the modification in place, but not those programs that 'poke' information into video memory and attempt to 'peek' it back later using video memory for RAM. Two early versions of Radio Shack programs used this technique, but have since been modified. Free updated programs are being made available to any customer with this problem.
- The company notes that programs created to run in upper case only should not be run with the lower case driver. This includes all existing Radio Shack business software. The lower case driver is included for the customer who wants to write his own programs in lower case.

Our verdict is that, in this instance, Radio Shack is generally in the right. . . and our reader somewhat out of line. While we certainly don't agree with all of Radio Shack's marketing practices, this seems to be pretty straightforward; our reader didn't have to go to court to get a refund. . .or express dissatisfaction.

When a reader asks for assistance, we're more than glad to help or intervene. But we can't solve problems that don't exist. If you encounter difficulty with a vendor, it is in your best interests to keep copies of all correspondence, advertising, literature and the like.

Give the vendor an honest chance to solve the problem. Remember, you may be the one who is wrong.

Be civil. It is difficult for an independent third party to assist you if you are filling the air with shouting, threats to sue, and slanderous remarks concerning business practices.

Finally, if you write asking for help with a problem, and the problem is resolved, let us know. We get involved in disputes as a service to our readers, but our first responsibility is to get a magazine published. We appreciate it when we don't have to waste time chasing down problems that have already been solved.

The TI Road Show

We were recently invited to spend a morning with Texas Instruments for a preview of a new (for them) form of marketing for the 99/4.

Basically, TI has sent out a number of mass mailings in the Los Angeles area inviting people to a free seminar explaining the basics of the home microcomputer, and a chance to try out the 99/4. TI calls it "hands-on experience."

TI isn't trying to reach hobbyists or businessmen with this traveling road show; it is trying to reach people who have had little or no exposure to the home computer system. Generally speaking, the people attending these seminars are older persons with a higher income and little or no knowledge of computers. Many are from technology-oriented career fields and are just somewhat interested in a computer for their home. A large number are women.

TI notes that many who come to the seminars have some rather misguided ideas about computers and just what the capabilities of a home system are. Many times they tell TI: "I am willing to spend X dollars for a computer than can do A, B, and C, but I'll be happy if it can do a lot more. . ."

Applications of major interest to the attendees generally are budgeting, tax recordkeeping, stock or investment analysis and the ability to manage more information at home than now possible.

Virtually none of the attendees want to learn how to program; they want modularized, menu-driven load-and-go software. □

Now! North Star Application Software!

North Star now offers application software for use on the HORIZON! Now you have one reliable source for both hardware and software needs! The first packages available are:

NorthWord—
NorthWord is a simple-to-operate word processing system designed for use with the popular North Star HORIZON. NorthWord enables you to increase office efficiency and cut document typing time and cost. NorthWord incorporates the most sought-after word processing features: easy editing, on-screen text formatting, simultaneous document printing, and much more. NorthWord can be integrated with other North Star software packages to produce customized letters, labels and reports quickly and efficiently.

MailManager—
MailManager enables you to compile and maintain complete organized mailing lists. Lists are easily accessible and can be compiled with a great deal of flexibility. Entries, corrections and deletions are easily made. The North Star MailManager can print your list on individual envelopes, on mailing labels, or in compact summary form.

InfoManager—
InfoManager is a powerful list-oriented, data management system. It will accept up to 50 categories of information for each record and has the ability to select and sort before printing. The North Star InfoManager has power and flexibility for many applications: product inquiry, inventory, customer/client records, calendar reminders, and as an easy way to fill in often-used forms.

GeneralLedger—
General Ledger and Financial Reporting, two programs in one, maintains general ledger accounts based on such input as checks, bank deposits and journal entries, and uses the information in the general ledger to produce customized financial statements and financial reports.

NorthWord is the central building block for all the North Star application software to follow. Packages now being tested include other accounting and professional application packages. For more information or a demonstration, contact your local North Star dealer.

CIRCLE INQUIRY NO. 47

NorthStar

North Star Computers, Inc.
1440 Fourth Street
Berkeley, CA 94710
(415) 527-6950
TWX/Telex 910-366-7001

LETTERS

Pebble-in-the-shoe

I'm a tax consultant. Thirty-nine of my clients are small businessmen with gross sales or revenues up to one and a half million. The universal pebble-in-the-shoe for this group is accounting needs. They are not large enough to afford an in-house accountant, yet outside accountants do not have the time to give them the attention they should have. Each client is asking me for a solution. They need mini capability at a micro price for a plug-in-and-run system. Such a system with its predesigned input formats and instructions, in the $10,000 to $15,000 range, would take off.

H. Craige DeMoss
Nashville, TN

Micros for cheating?

Re "The Teacher and the Personal Computer. . ." (IA Jun 80), I have spent much time trying to use computers in similar ways in my classes. I was greatly disturbed, however, at Samuel Spero's suggestion that computers be used to fake data on physics labs. When I learned physics, I would have been immediately expelled for making up lab data. The only way progress is made in experimental physics is when experiments do not turn out as expected. Working with mock data violates every principle of laboratory instruction.

Thomas R. Davis
Menlo Park, CA

Get with the programs

If the computer is to take a place in the home on the level of most other modern conveniences, it will have to be programmed by the user. To do this, there will have to be a number of sources for information pertaining to programming. There is a great void in this area. The available information only scratches the surface of what can be done with a computer.

For example, I have eight or 10 books on Basic. In one of my programs, I wanted to use a 'print using' statement that would print: 'The total value of the property is $35,900.' Not one book showed how this could be printed. All the books showed was how to print a column of figures.

Another example: How do you use two data statements in the same program and have only the first statement read, then later in the program have only the second statement read. It is not covered in any manual or reference material that I can find.

There are unlimited programming problems that your publication could address that would be helpful to the home and small business computer owner.

Helping teach programming and computer languages, serving as a watchdog of unethical practices relating to the computer industry, and testing software programs to determine whether they perform as advertised would be a service worthy of your magazine.

B. H. Martin
Richmond, IN

See this month's Editor's Notebook.

Just what the doctor ordered

Whenever I've had a medical problem, the physician's response has been "I'm not sure what is ailing you, but we should try to find out". . .for a fee. In most cases, the problem was never solved and often made worse. I was therefore amused by a statement from James K. Robinson, MD (Physician's Approach. . .", IA Jun 80). He expressed disdain for ". . .hardware and software vendors who could not, by the widest reaches of the imagination, meet our needs, but were willing to try. . .for a fee."

Apparently what's good for the goose is not good for the gander.

Wil Schuemann
Parkersburg, WV

Readers to the rescue?

Has anyone written a program to interface a Summagraphics Bit Pad One to a Southwest Technical Products 6800 or 6809? The Summagraphics Bit Pad has serial output with the format 'XXXX,YYYY',CR-LF (strappable internally for no LF).

The program can be in Basic or in machine code. The values of the bit pad should be capable of diskette storage so that this saved data can be read back by another program to drive a Houston Digital Plotter (Hi-plot). The programs may run independently, or, if really smart—the bit pad on one port on an MPS card, and the Hi-plot on another card on another port.

Dr. Denis Saunders
343 Oak Ave., Ferndale,
Randburg, 2194, Rep. of South Africa

I have found myself in a predicament, after purchasing my home computer, VideoBrain manufactured by Umtech. The computer uses encoded programs written on cartridges using APL/S, a subscript of full APL.

When I purchased the computer, I did not get an APL/S cartridge to enable me to write my own programs, as most cartridges are dedicated programs. Now that I want to write my own, I have found that the company has gone out of business.

I have tried in vain to find anyone who still stocks the cartridge. I have even tried to get the instructions for the APL/S program and have my own Eprom programmed to make my own cartridge.

I would be very grateful for any assistance.

Semper F. Dick
102 Benshire Dr.
Scarborough, Ont., Canada M1H 1M5

Inclusion of minis?

As a minicomputer enthusiast and owner of a DEC 8/I 12-bit mini, may I suggest a focus on older systems in your publication? There are hundreds of surplus DEC, Data-General, Nova and other minis available that are attractive to hobbyists.

Jeffrey L. Flaws
Matteson, IL

Help for the stranded

I am an owner of an Interact computer who found out that the manufacturer went bankrupt last summer. I feared my computer would be worthless when my cassette tapes wore out.

By chance I found a strong user's club which put me in touch with sources for new and replacement tapes. Your readers can write Interactions, c/o Stephen Cook, 15356 Prevost, Detroit, MI 48227 for information.

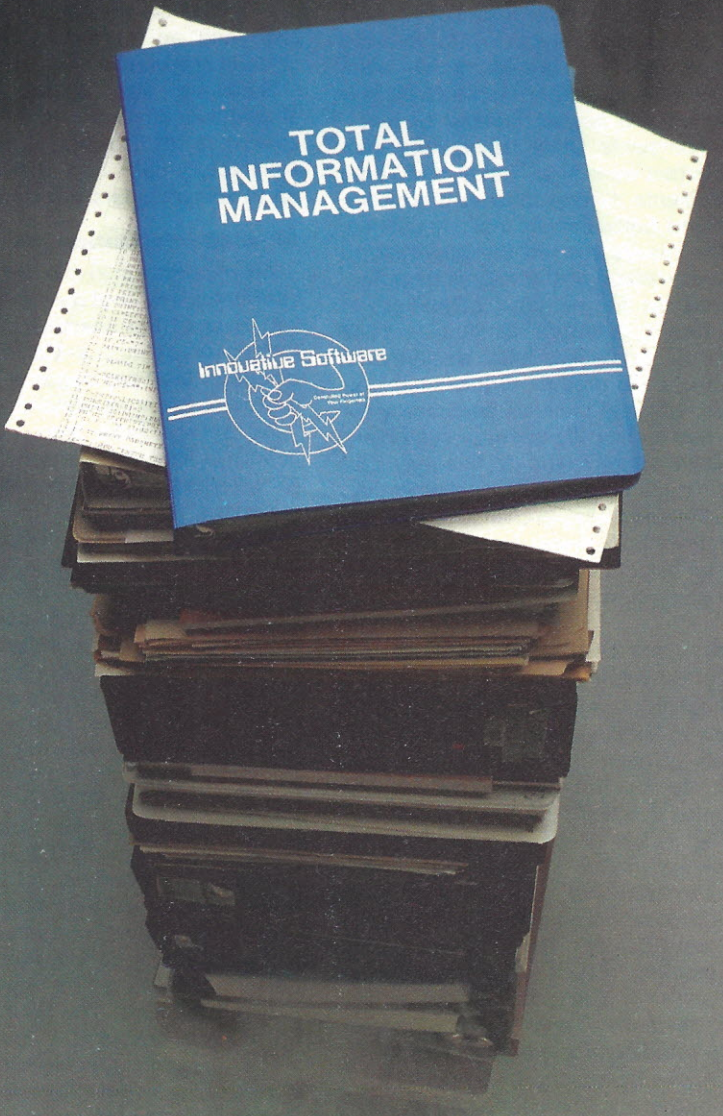
David Ross
Pittsburgh, PA

Ooops. . .

Re *Parts Inventory Control,* Jul 80, pg. 56, one line of the program was inadvertently scrambled; it should read:

```
320 DOL=S*R:YDOL=YT*R:DT=DT+DOL:YTDOL=YTDOL+YDOL
PRINT I;TAB(10);P$;TAB(20);D$;TAB(40);S;" $";DOL;TAB(60);YT;" $";YDOL
GOTO 310
```

YOU'VE LOOKED AT THE REST NOW LOOK AT THE BEST!



INTRODUCING T.I.M.

You've decided that a computer might help you perform certain functions in your business more efficiently. Now you are looking at different machines and software. Here's where your problem starts.

You can compare disks against disks and printers against printers. But when it comes to software there are so many programs on the market, it's like comparing apples to oranges. If you buy the right one you win. If you don't, you lose.

That's where we come in. To help you win. We're Innovative Software and we've just finished developing and field-testing a new program for the micro-computer market. It's called T.I.M., Total Information Management. We've been told that it's one of the most comprehensive data base management programs on the market. Check out these features and see if you agree:

- Menu-driven and user-oriented. Extensive computer expertise is not needed.
- Contains 4 user libraries. Any number of file formats, report formats, list formats, and record selection formats may be stored.
- Contains a unique text-editor that may incorporate field information from the data base to write checks, invoices, form letters, etc.
- Eliminates resorting of data files by automatically merging new or updated records back into the host file.
- Supplied with a 125-page reference manual with over 30 pages of examples.
- Available under Micropolis BASIC and CP/M* MBASIC**

See how T.I.M. can help you win in the software game. If your dealer doesn't have T.I.M. yet, give us a call or write to the address below.

Innovative Software
Computing Power at Your Fingertips

9805 Holly
Kansas City, Mo. 64114
(816) 942-8778

*Trademark of Digital Research
**Trademark of Microsoft

UPDATE

Stores say simple computerese would trigger micro sales

Computer retailers want simpler-to-use equipment, manuals and software, according to a recent survey for beginners. The stores answering the ten-question survey said their sales would increase significantly if they had more products suited to nontechnically-oriented beginners (persons with no more than high school math and physics).

Over half the prospective customers appear to possess such a background. At present, only about half of the equipment, software, manuals and promotional literature is suitable for these nontechnically oriented customers who are beginning to show sparks of interest.

The stores were also invited to name companies and brands most commented on positively. Apple Computer received the largest number of favorable comments. Other manufacturers somentioned were Atari, Tandy, and Texas Instruments. Among software producers, Personal Software was most mentioned.

The results support the view that, to take advantage of the enormous potential market for personal computers in the 1980's, hardware manufacturers and software producers must design their products and prepare promotional literature, ads and manuals specifically for the non-technically oriented customer.

Manufacturers who do this may capture a major portion of the consumer market. Those who fail to change may be left with the infinitesimally smaller market of the technically educated who can speak and understand "computerese."

Computers cost less, but expert questions value

If airlines had been able to match the gains in efficiency and reduced price levels of computers, it would be possible to fly from New York to Los Angeles in a half hour for 25 cents.

That is the blunt example that computer expert Don Simmons uses to tell companies about how far electronic data processing equipment has come in the last 20 years. His firm, Simmons & Associates, Chicago, assists companies to decide which computer fits their needs best, and helps design software programs for maximum results.

"A lot of companies sharing time on computers would be better off to put in their own equipment because of the way the price has come down," Simmons said. "The front-end costs will be heavy, but companies can devise their own programs and change them or add to them without having to rely on software packages that might not fit their needs."

However, companies that decide to invest in a computer often get oversold, Simmons believes.

"They aren't knowledgeable about the different models, and buy more capacity than they need or will need anytime in the foreseeable future," he said.

Another problem first-time buyers of computers may find is that delivery on popular models can take 12-14 months.

"If a computer system can save a company $100,000 a year—as they often can—it is important to get the equipment without a long delay," he said.

Once the computer is installed, packaged programs can be plugged in so the equipment performs at once.

"Finding the right package is important, and so is training the staff to operate the equipment," he said.

Simmons also thinks companies should take an audit of their computers to see if the equipment is providing satisfactory data, and whether it is set up to analyze costs effectively.

AP adds information retrieval

An experiment in the technology of information retrieval involving a group of members of the Associated Press and Compuserve Inc., a Columbus, Ohio computer firm, has begun to offer information to home personal computers.

AP serves more than 1,370 newspapers in the U.S. The newspapers and AP will provide their information to Compuserve's computers in Columbus. Anyone with a personal computer will be able to obtain this information by dialing special telephone numbers. Users pay $5 an hour to access news, sports, business and feature data provided by the newspapers and AP.

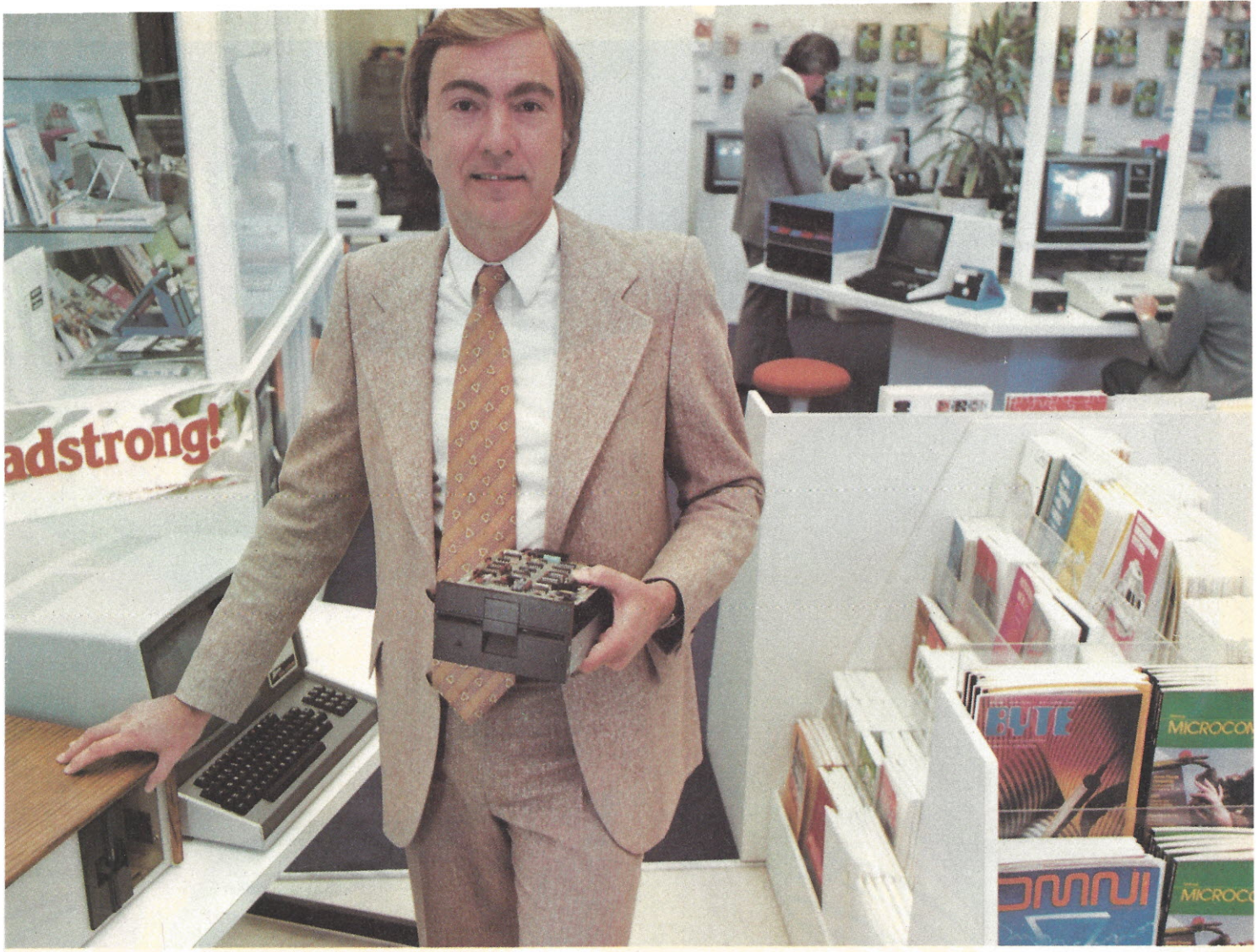
The first newspaper to participate will be the *Columbus Dispatch* in Ohio. Each newspaper will experiment for six months.

Late word. . .

IA has learned that Source Telecomputing Corp. (STC) of McLean, VA and Tymeshare Inc. of Cupertino, CA, have signed a development and pilot operations agreement under which Tymeshare will provide facilities and services to increase the user capacity of The Source.

Sources inside STC indicate that non-prime computer time is being purchased from Tymeshare and that the company will consult with TCA as to future operating locations, systems acquisition and software development.

Industry observers feel that this is but one in a series of actions by TCA to beef up their financial and operational capabilities.



"For reliable data storage, you can't beat Shugart's Minifloppy."™

Raymond Schlitzer, Owner—
Computerland, San Francisco

"I sell systems my customers can depend on. That's why most of the personal and small business computer systems sold here feature Minifloppy disk drives. I know from experience I can rely on the Minifloppy."

Since 1976 Shugart's Minifloppy has been used by more small computer system manufacturers than any other drive. In fact, more than half-a-million Minifloppys have been installed. The Minifloppy looks small—but it stores a lot of data. 250 kilobytes on one side, or up to 500 kilobytes in the double-sided model. That's about 50 pages of printed information on a single-sided Minidiskette, and twice that on the double-sided version. You'll have plenty of storage capacity for your programs, letters, forms, or ledger entries. And you find your data fast, too, because the Minifloppy is a random access device that eliminates the need to search for your data serially as you must with a tape cassette unit.

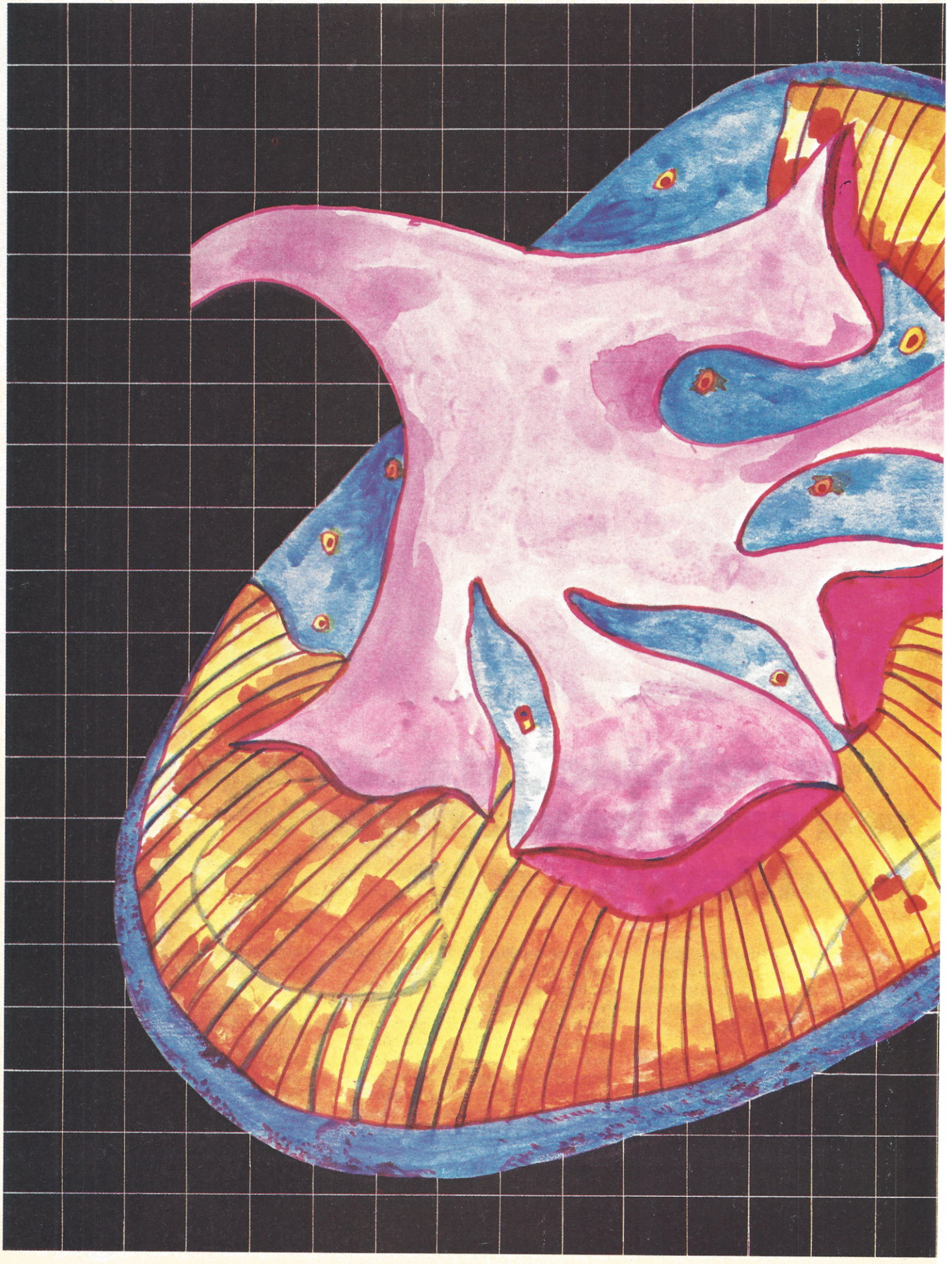
No matter what problem you're solving with your computer system, you can rely on Shugart's Minifloppy for data storage. We're known as the Headstrong company for good reason. We're Headstrong about reliability, quality, and value. Ask your dealer. He knows us.

Rely on the Headstrong Company.

Shugart®

TM—Minifloppy is a trademark of Shugart Associates.

475 Oakmead Parkway, Sunnyvale, California 94086



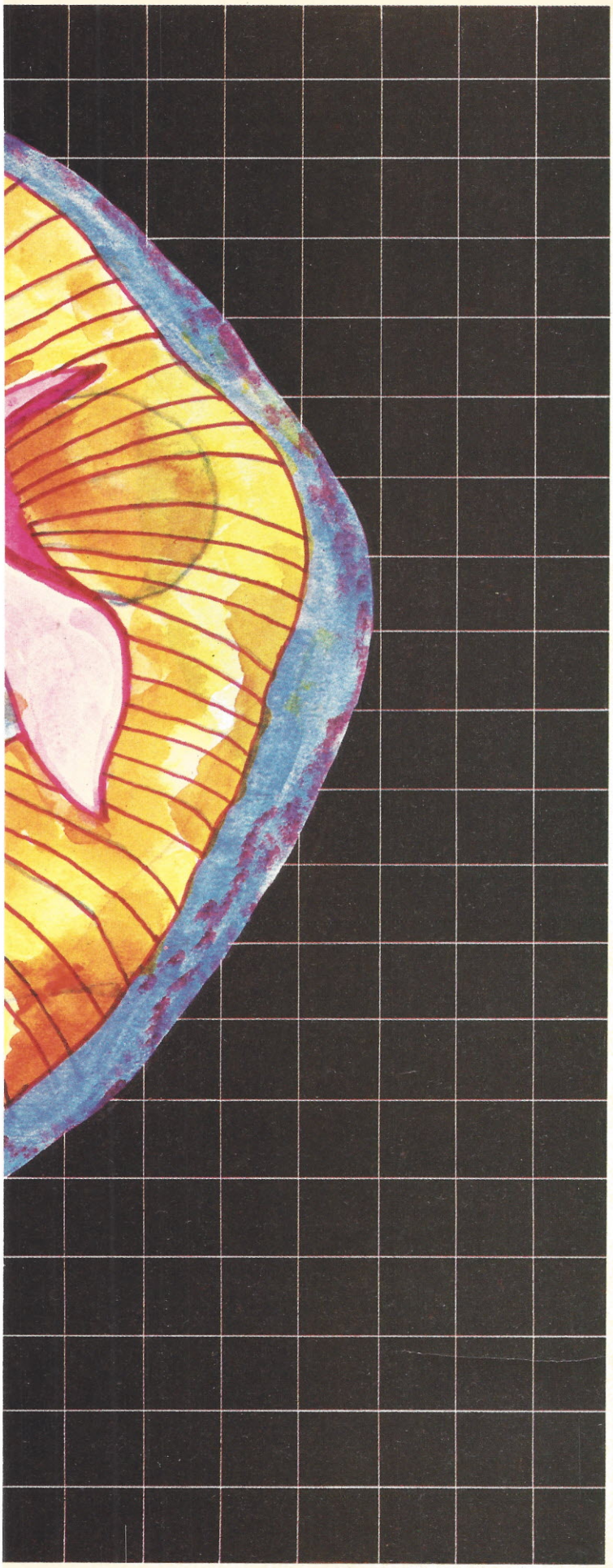


Illustration by Samantha Lee

LUNG:
A Program to Calculate Spirometric Data

by James K. Robinson, M.D.

A microcomputer can be used to save time and avoid errors in calculating the data generated by the performance of a screening pulmonary function test (spirogram) in the average primary-care physician's office. The purchase of an independent microcomputer to perform this task offers versatility as well as a savings compared to the purchase of a so-called ''computerized'' pulmonary function device (spirometer).

A program, written by the author in Microsoft Basic for an Exidy Sorcerer, accepts raw data from the spirograph and converts flow values from ambient atmospheric temperature and barometric pressure conditions (ATPS) to body temperature and pressure conditions (BTPS). Volume and flow normal values are then calculated based on subject's height, age, and sex, and compared against actual values and expressed in percent of expected. These results are displayed on a chart suitable for printout if a hard copy printer is available, or can be hand copied directly from the CRT to the patient's record.

Continued on Page 122

PEARL™ follows tradition

PEARL builds on the legacy of master craftsmanship which has produced so many of America's finest achievements — by establishing a new standard for excellence, for availability, for innovation in microcomputer software.

PEARL (Producing Error-free Automatic Rapid Logic) is a program which generates new programs. Available in versions for nonprogrammers as well as computer professionals, PEARL is designed to run under CP/M* on any microcomputer with at least 48K.

PEARL lets the end-user define and produce automatically new source code for all new applications software — in BASIC — in a matter of hours. PEARL's modular, structured code provides a classic teaching tool for software development staffs, yet on-screen menus and prompts make it easy for nonprogrammers to use.

Tradition and excellence aren't just things available yesterday. PEARL brings them to you today. You can generate error-free microcomputer software NOW. Call your nearest dealer today.

Manuals	$ 25.	
Level 1	130.	(for non-programmers)
Level 2	350.	(for those with programming skills)
Level 3	650.	(for software developers)

evolutionary software for the 1980's

Computer Pathways Unlimited, Inc.
2151 Davcor Street S.E. • Salem, Oregon 97302
(503) 363-8929

CIRCLE INQUIRY NO. 10

*CP/M is a trademark of Digital Research

Jurisprudent Computerist

By Elliott MacLennan
Attorney at Law

The Computer Software Protection Act

Any legal attempt to address the issue of computer software protection immediately causes two veteran enemies to resurface and prepare for enduring combat. The first force is the desirable social interest in encouraging the continuing development and free promulgation of state-of-the-art program information. Sparring against this is the economic interest, namely, legal insurance to the program developer that his economic investment in the programs will not be misappropriated.

To equitably resolve the competitive forces of economic rights vs. social interest, Congress and the courts have traditionally relied upon three protective systems: trade secret protection, patent coverage, and copyrights.

I will discuss where these three systems lack fail-safe features and, concurrently, discuss what steps are being taken to bridge the gap occasioned by technology making these traditional systems obsolete.

Gaining a competitive edge

Trade Secrets. Protection of proprietary or ownership interests in software is the oldest and most frequently used system. Trade secret laws come from our common law heritage. They protect a businessman's ideas, thereby giving him an advantage over competitors who lack such knowledge. Trade secrets are legally protectable as long as they remain secret. Herein lies their flaw. Maintaining secrecy is difficult when a businessman must mass-market his product. More importantly, keeping a secret a secret under these market conditions requires extensive monitoring. That's expensive.

Trade secrecy laws thus protect a businessman's economic rights. However, their very existence flies directly in the face of the social interest by squelching the technological exchange of ideas.

Patents. Do not require extensive monitoring; gives its holder a limited monopoly as a reward for tangible industrial innovation. Patents provide the most effective legal protection. In exchange, a software developer is encouraged to make full program disclosure to the public. Hence, the economic and social forces are evenly balanced.

Regretfully, patentability is not generally available for software. The little protection available is woefully deficient. And even if such protection was readily available, the issuance of a patent is often costly and always a lengthy process. The patent office is not known for its speed; a technologically innovative program approach can become a relic by the time a patent issues. On a positive note, however, the income tax treatment of patents is more favorable to the taxpayer-holder than that afforded trade secrets or copyrights.

Protecting physical embodiments

Copyrights. Quick, simple, dirt cheap to obtain and require little monitoring. Here the good news stops. Copyrights protect expression. More specifically, they protect the physical embodiment of the program, not the underlying algorithm. Copyrights advance the social interest but retard the economic rights of the software developer because of limited protection.

Sometimes even the program itself is not protected. Nowhere was this more strikingly displayed than in the Data Cash Systems, Inc., vs. JS & A Group, Inc. (451 Patent, Trademark & Copyright Journal E-1, Oct 25 1979). In a classic case of one developer suing another "developer" for misappropriation of proprietary interest in a ROM object program, the judge held the copyright laws did not apply because an object program cannot be read by the human eye (i.e., without special equipment). Therefore, Data Cash's program for a computerized chess game was not afforded protection.

How did this sorry state of affairs come about? In 1964, Congress, seeking a classification for computer software, categorized it as "literary work," which, by definition, must be capable of being read by the unaided eye. Applications software, for example, is copyrightable under this theory; object programs are not.

What have our legislators done to resolve this fundamental inequity in the projected $70 billion software market of 1980? Congress established a National Commission on New Technological Uses of Copyrighted Works in 1974. Three years later, the Congressional software subcommittee report recommended further distortion of the copyright law by making "minor" changes to protect computer software. Didn't someone once say that a camel was a horse put together by a Congressional subcommittee?

What is needed is special legislation protecting software, not further distortion of legislation that already affords scant protection. The proposed Computer Software Protection Act may be such legislation.

Curiously, the National Commission's draft was never introduced into Congress. Why not deal with a technologically innovative concept by giving it its own legal "space" and concomitant protection? The subcommittee's manner of making a new change to archaic legislation for protection of an innovative idea has created MacLennan's Law: Congress' ability to adequately protect technological innovation decreases in direct proportion to increases in computer speed and software sophistication. □

AL BAKER'S GAME CORNER

Having Fun. . .Seriously

Last month we explored the use of computers as a tool to simulate reality and have fun doing it. I didn't label the program "educational" since a lot of people don't have fun if something is "educational." Now that you were tricked into playing with an educational program, I'm coming out into the open. The program this month is educational.

My seven-year-old daughter, Jennifer, and I often work on projects together. Not long ago, it was curves: parabolas, circles, hyperbolas, and ellipses. Except for circles, these are all pretty hard for a seven-year-old to grasp. It took about 20 minutes to write the program Parabolas.

After spending half an hour making parabolas with needle and thread, I sat her down at the Apple. For the next hour, she was enthralled at making parabolas happen on the TV screen. I'm sure she doesn't remember the equation for a parabola, but she can recognize and draw one.

Since then, we have worked with other geometric curves and had a lot of fun. What's important is that she is comfortable with shapes, curves and graphs. Many of the things her fellow students will find strange, or even scary, will be old friends to her.

Parabolas

This program uses Applesoft and the high resolution graphics screen to draw parabolic curves. The equation for a parabola is

$$Y = \frac{X^2}{4F}$$

where F is the focal point of the parabola. This equation, in modified form, is on line 50 of the program. The function FNP(X) returns the value Y converted into an actual plot address on the TV screen. The graph is 14 units across with the X axis at the bottom of the screen (figure 1). Each dot is .05 units across giving the Apple high resolution horizontal value of 14/.05 = 280. The origin is at the bottom center of the screen, at location 139,159. Since each dot is .05 units high, a vertical distance of 1 unit is 20 dots. Therefore to convert a value of Y to a dot coordinate is 159–20*Y.

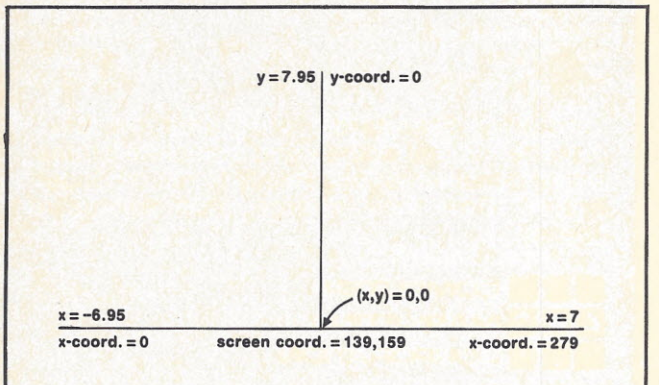


Figure 1. Parabolic curve graph paper.

Line 90 positions the next print statement at the bottom of the screen. Lines 100 through 120 create the high resolution graph paper. The remainder of the program is the plotting loop. Line 230 plots the focal point if it appears on the screen. Lines 270 through 300 determine the starting and ending values of X while plotting. This is variable L. First, we assume that X can span the entire screen from −6.95 to + 6.95 (points 1 to 279). If Y won't fit on the screen, we pick a maximum Y value of 7.9 (vertical position 2) and use the inverse parabolic function X = 2*SQR(Y*F) to determine the limits of X. Lines 340 to 370 then plot the curve.

Ellipses

Here is another program Jennifer and I have fun with. This one has a lot more play value than Parabolas. With Parabolas we made different kinds of tulips and other flowers. With Ellipses, we made the CBS eye, funny shrinking footballs and baseballs, and cotangent ellipses within circles within ellipses within. . .

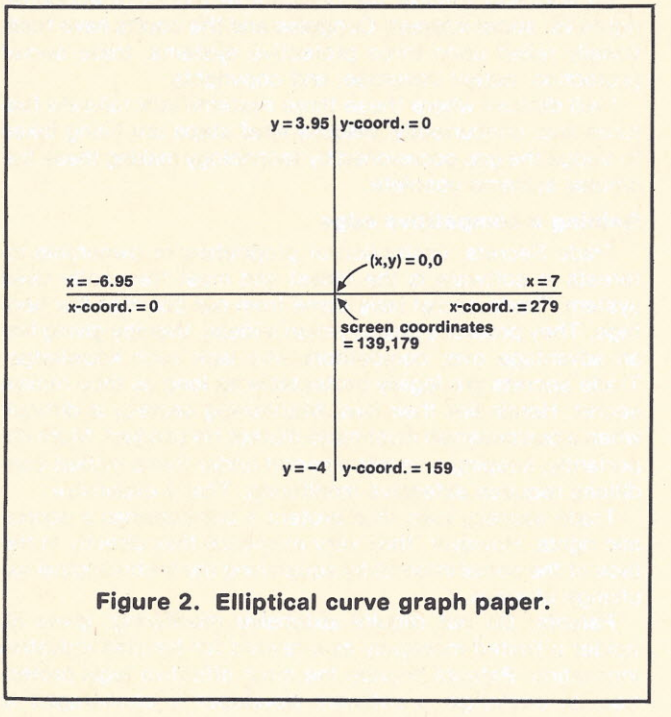


Figure 2. Elliptical curve graph paper.

Figure 2 is the graph paper we put on the screen for Ellipses. Here the origin is in the center. Line 50 is again the formula function. Since it must be used both above and below the origin, FNE(X) converts Y to a dot address but doesn't add or subtract it from the Y-origin value of 79.

Lines 230 and 240 verify that the ellipse will fit on the screen and lines 250 through 280 draw the marks on the axes which the curve will move through. Finally, lines 320 to 350 draw the top half of the curve and lines 360 through 390 draw the bottom half.

We've only touched the ways the computer can be used to show people the beauty of mathematics. Try coming up with programs on your computer to do other curves. How about a hyperbola or a trigonometric function such as SIN(X)?

If you didn't find fiddling with curves as much fun as most seven-year-olds will, wait till next month. We are going to play an exciting guessing game. (Just don't tell anybody its educational. Certainly, don't tell anybody it's about history. Yech!)□

Program follows

MicroNET is just the tip of the iceberg

We've been telling you that MicroNET, CompuServe's personal computing service, is the best thing that's happened to personal computers since electricity. It still is, but now there's more. A lot more.

Welcome to CompuServe's information service.

- *News. Weather. Sports.* Major regional newspapers. Plus international news services.
- *Finance.* MicroQuote. Updates and historical information on stocks, bonds and commodities.
- *Entertainment.* Theatre, book, movie and restaurant reviews. Plus opera, symphony, ballet, dance, museums, galleries...
- *Electronic Mail.* Create, edit, send and receive messages from any other CompuServe user...nationwide.
- *Home & Educational Reference Service.* Anything you want to know...from encyclopedia information to household tips.
- *CompuServe user information.* In case you need technical help...and information on new services as they become available.
- *MicroNET.* All we've offered before and added lately with more to come. This includes Software Exchange,line printer art gallery, challenging games, programming languages, word processing, business & educational programs...and much, much more.

So we're raising the price. Right?

Wrong! All you pay is a small hook-up charge, and $5.00 per hour billed in minutes to your charge card. You need a 300 baud modem and we're a local phone call in more than 200 North American cities.

Write for information. This is almost too good to believe, but we're delivering as promised.

CompuServe

Information Service Division
5000 Arlington Centre Blvd.
Columbus, Ohio 43220
(614) 457-8600

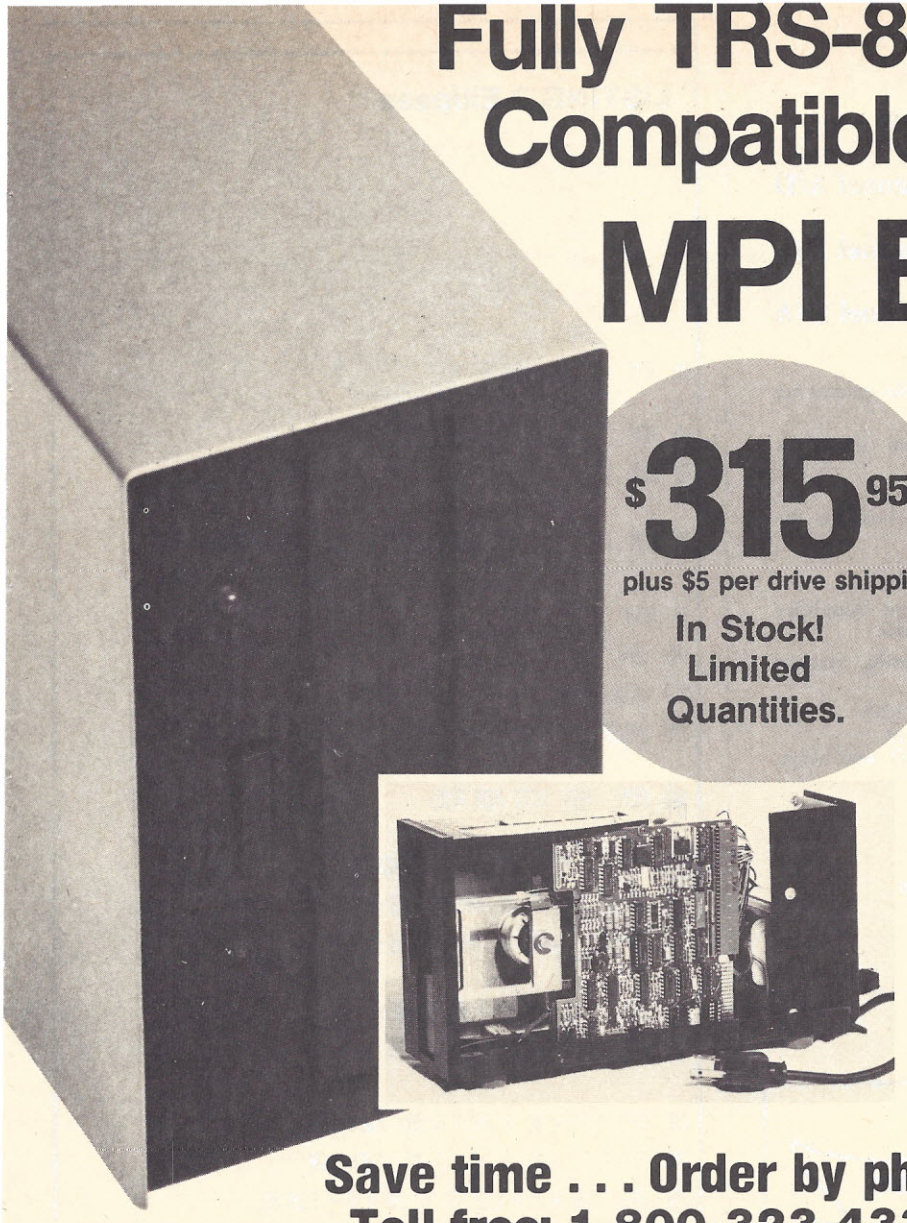
LISTING 1 Parabolas

```
10 REM     PARABOLAS

40 REM  THE PARABOLIC EQUATION:
50 DEF  FN P(X) = 159 - 20 * (X * X / (4 * F))

70 REM  PRINT AT BOTTOM AND SET UP THE GRAPH PAPER

90 POKE 37,23
100 HGR
110 HCOLOR= 3
120 HPLOT 139,0 TO 139,159 TO 0,159 TO 279,159

140 REM  ASK JIJI FOR FOCAL POINT

160 PRINT : PRINT : PRINT "JENNIFER,"
170 INPUT "  WHERE IS THE FOCAL POINT? ";F

190 REM  USE POSITIVE FOCAL POINTS ONLY
200 REM  PLOT FOCAL POINT IF ON SCREEN

220 IF F < 0 THEN 160
230 IF F * 20 < 159 THEN  HPLOT 136,159 - F * 20 TO 142,159 - F * 20


250 REM  LIMIT PLOTTING TO POINTS ON SCREEN

270 Y =  FN P(7)
280 L = 6.95
290  IF Y <>  = 0 THEN 340
300 L =  SQR (7.9 * F) * 2

320 REM  PLOT THE GRAPH

340 FOR X = L TO  - L STEP  - L / 140
350 XN = X * 20 + 139:YN =  FN P(X)
360  HPLOT XN,YN
370 NEXT X

390 REM  LET'S DO IT AGAIN

410 GOTO 160
```

Fully TRS-80® Compatible

MPI B/51 DISK DRIVE

FREE CATALOG
Hundreds of other computer products at major savings. Cut out this coupon and send with your name and address.



$315.95
plus $5 per drive shipping
In Stock! Limited Quantities.

Includes: Case and Power Supply

Fully tested
Guaranteed for 90 days!

Here's why the MPI/B-51 is the drive for success:

- 40 tracks
- 5 ms track-to-track
- Auto-eject
- Hi-Temp stability
- Fully-closable door
- Speed constant <1½%
- Double density head
- Optical sensors—no switches
- 102K per disk

ADDS MORE POWER TO YOUR SYSTEM

TRS-80™ TANDY CORP.

Save time . . . Order by phone
Toll free: 1-800-323-4335
IN ILLINOIS CALL: 312-251-5955

Other Money Savings Opportunities
Order by Phone or Mail

Cables
- 2 drive $34.95
- 4 drive 44.95

Diskettes
- Verbatim ..... 10 for $29.95
- Memorex ..... 10 for $26.95
- Georgia Magnetics ... 10 for $28.95
- Dysan ..... 5 for $24.95
- Plastic file box ..... $3.95 each

Operating Systems
- TRS DOS 2.2 ..... $14.95
- Percom Patch #4 ..... $9.95
- New DOS 40 track .. $49.95
- New DOS+ 40 track .. $99.95
- TRS DOS manual $5.95

TRS-80
- 16K level II with keypad $749.95
- Expansion interface ØK $269.95

Printers
- Centronics 779-2 .. $1,050.00
- Centronics 730 ..... $899.00
- IDS-440 ..... $949.00
- NEW! TX-80 ..... $945.00
- NEC Spinwriter 5530 ..... $2,679.00

16K Memory Kits
- 300ns $69.95
- 250ns $74.95
- 200ns $84.95
- MPI Service Manual $3.00
- MPI Engineering Manual $30.00

MIDWEST COMPUTER PERIPHERALS
P.O. BOX 437 • WILMETTE, ILLINOIS 60091

CIRCLE INQUIRY NO. 44

Midwest Computer Peripherals

P.O. Box 437 • Wilmette, Illinois 60091

WE HONOR master charge THE INTERBANK CARD VISA ACCEPTED!

085

Quantity	Description	$ each	Total

6% Ill. Tax
Shipping ($2.50 min.)
TOTAL
(Min. Order $10.00)

☐ Check enclosed
Bill my ☐ Visa
☐ Master Charge
Acc. No. _____ Exp. _____

☐ Please send catalog

Name _____
Address _____
City _____ State _____ Zip _____

LISTING 2 Ellipses

```
10 REM       ELLIPSES

40 REM    THE ELLIPTIC EQUATION:
50 DEF  FN E(X) = 20 * B *  SQR (1 - X * X / (A * A))

70 REM  PRINT AT BOTTOM AND SET UP THE GRAPH PAPER.

90 POKE 37,23
100 HGR
110 HCOLOR= 3
120 HPLOT 139,0 TO 139,159 TO 139,79 TO 0,79 TO 279,79

140 REM  ASK JIJI FOR AXES

160 PRINT : PRINT : PRINT "JENNIFER,"
170 INPUT "  WHAT IS THE X SEMI-AXIS? ";A
180 INPUT "  WHAT IS THE Y SEMI-AXIS? ";B

200 REM  USE POSITIVE AXES ONLY
210 REM  PLOT AXES IF ON SCREEN

230 IF (A <  = 0) + (A > 6.95) THEN 160
240 IF (B <  = 0) + (B > 3.95) THEN 160
250 HPLOT 20 * A + 139,73 TO 20 * A + 139,85
260 HPLOT 139 - A * 20,73 TO 139 - A * 20,85
270 HPLOT 133,79 - 20 * B TO 142,79 - 20 * B
280 HPLOT 133,79 + 20 * B TO 142,79 + 20 * B

300 REM  PLOT THE GRAPH

320 FOR X =  - A TO A STEP A / 70
330 XN = X * 20 + 139:YN = 79 -  FN E(X)
340 HPLOT XN,YN
350 NEXT X
360 FOR X = A TO  - A STEP  - A / 70
370 XN = X * 20 + 139:YN = 79 +  FN E(X)
380 HPLOT XN,YN
390 NEXT X

410 REM  LET'S DO IT AGAIN

430 GOTO 160
```

There is a magazine on microcomputer techniques important enough to subscribe to even when written in the German language: CHIP.

You have to read it when you want to know what's going on in Europe.

CHIP is the leading magazine on microcomputer techniques in Germany. CHIP specializes on everything important in Germany and Europe. Background stories, research and practical appliances, testing, opinions and facts. Everything that counts. CHIP is instructive, enjoyable and informative, even when reading the German language is a little bit difficult for you.

Subscribing to CHIP means you have a hot line into microcomputing in Europe. You get your CHIP by airmail or surface mail.

Subscriptions to CHIP Magazine c/o Data Dynamics Technology, 16704 Marquardt, P.O. Box 1217, Cerritos, CA 90701, U.S.A.

YES, I want to subscribe to CHIP for one year (12 issues).

☐ Airmail $50

☐ Surface mail $32 (please allow 6 to 8 weeks for delivery)

☐ Check enclosed (payable to Data Dynamics Technology, 16704 Marquardt, P.O. Box 1217, Cerritos, CA 90701, U.S.A.)

Allow 8-12 weeks for first issue

Date _____

Signature _____

Delivery address _____

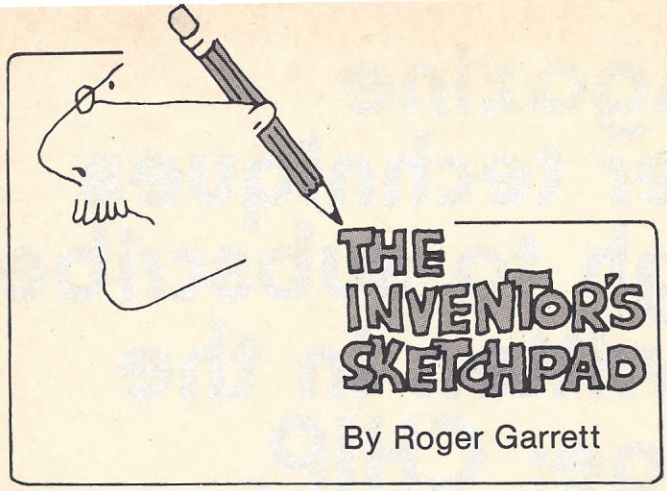
Name _____

Profession _____

Street/No. _____

City _____

State _____ Zip _____



Automatic Acquisition of 3-D Information

One approach to the problem of extracting three-dimensional information from images (IA Aug 80) is to experiment by trying out modifications, but without adding expensive equipment, most notably the television camera and frame grabber.

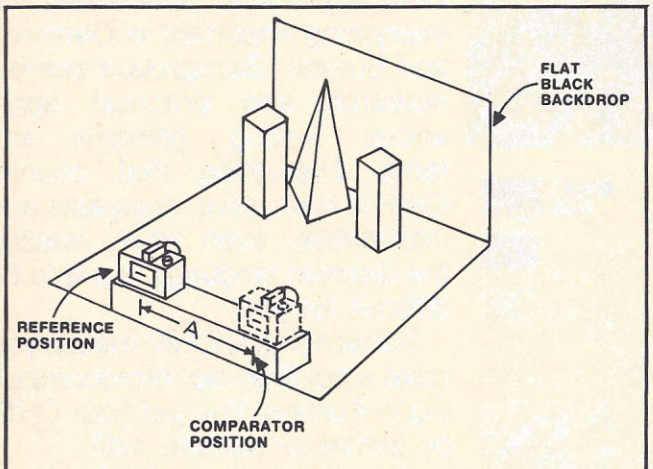


Figure 1. Experimental setup.

Instead of a TV camera, use a standard camera. It is best to use a good quality 35mm rather than the small pocket type, since we want good resolution. Set up a flat black background with several relatively simple objects such as boxes, balls and pyramids (figure 1). Progress to more complicated objects after the first few experiments.

Set up the camera so it faces the objects and is focused on them. The centerline of the camera should be perpendicular

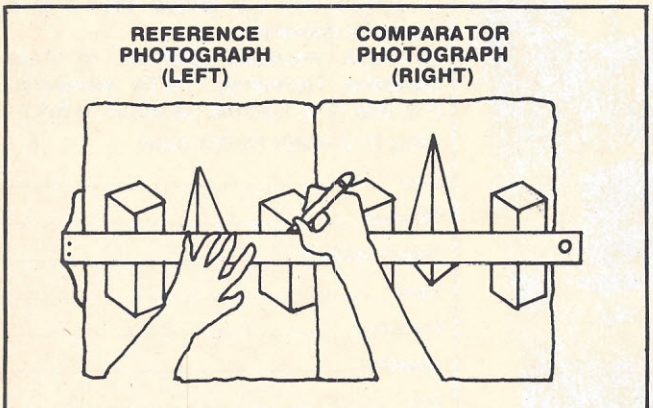


Figure 2. Comparison of photo images.

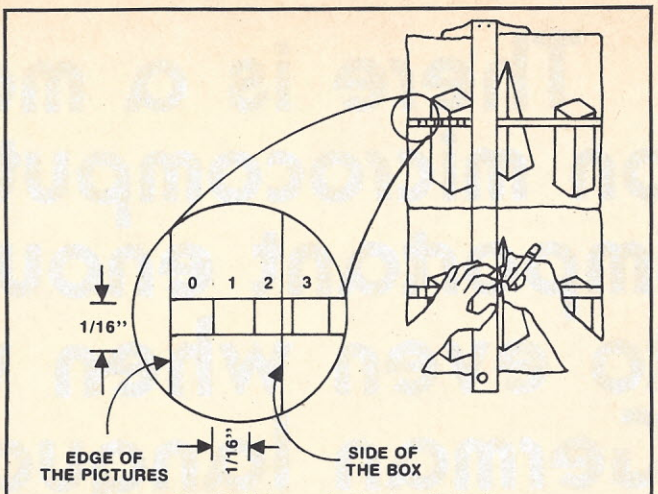


Figure 3. Division of photos into boxes.

to the backdrop. Place the camera in the reference position shown, take a picture, move it to the comparator position (being sure to keep it at the same height and perpendicular to the backdrop) and take another picture. Do not change the lighting between pictures, since it is important that the illumination and shadows remain the same.

After you have the pictures developed and enlarged to at least 5 x 7, set them side by side (figure 2) so the two (nearly identical) images are parallel. With a straight edge, draw two lines through some portion of the picture which intersects several objects. The lines should be about ⅛ to ¼ inch apart.

Place the photos one above the other and, using the straight edge, divide the parallel line in each photo into a set of boxes (figure 3). It is of utmost importance that each box in the reference photo have a correspondingly positioned box in the comparator photo. You can do this by using a T-square as shown and making sure that the two photos don't move while you draw the lines. Taping the photos to a drawing board or table top can help. After all boxes have been drawn, number them as shown, in both the reference and comparator photos.

Now that we have divided the selected "scan" line in each photo into pixels, we need some way to digitize the scan lines. We need a grey-scale chart (figure 4), and can make our own using the scale in figure 4a. Mount the scale on a cardboard backing and divide it into equal sections as shown here.

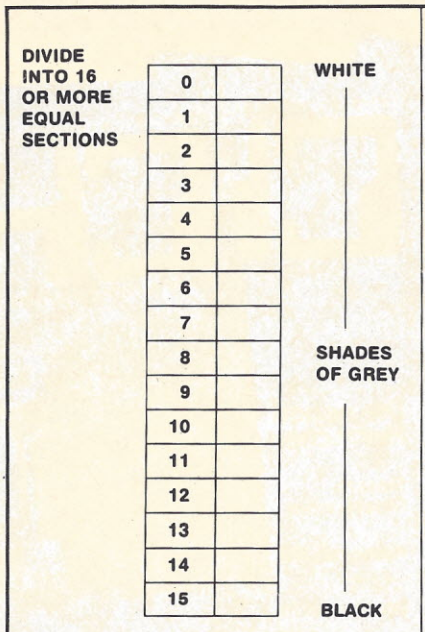


Figure 4. Standard grey scale chart.

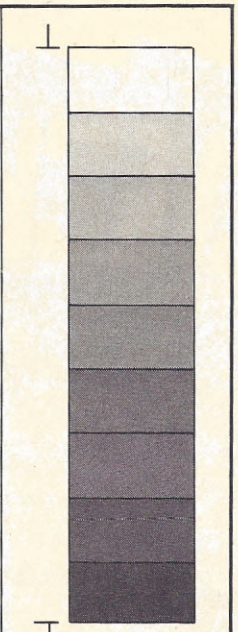


Figure 4a. Grey scale

More Sonic Echos

The saga of the sonic transducer continues. I'm still getting inquiries for the one described in this column. I have now learned that Polaroid sells a sonic transducer used in its Solar Land camera to anyone interested in experimenting in non-camera applications. A designer's kit is available for $125 from: Polaroid Corp., Ultrasonic Ranging Marketing, 20 Ames St., Cambridge, MA 02139.

You can use 16 divisions or any you choose. Try different ones during your experimentation to determine the optimum. Sixteen should be good for starters.

Now comes the tedious chore of digitizing the scan lines for each photo. It might be helpful to make a chart for each photo, numbered from zero to the number of boxes in the pictures with a space next to each box number. This is where you will write in the digitized value.

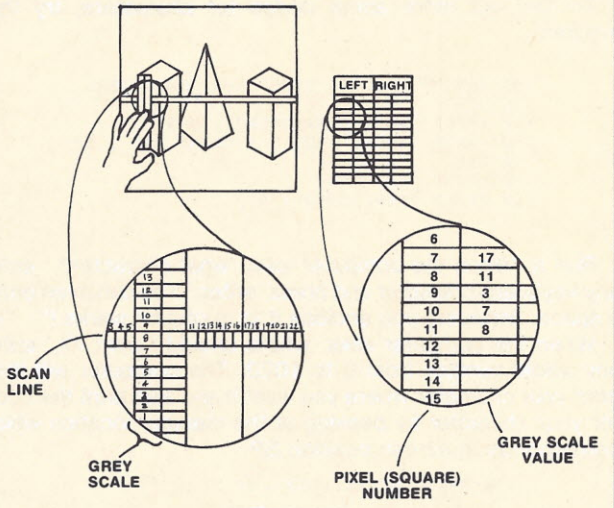


Figure 5. Comparison of photo with scale.

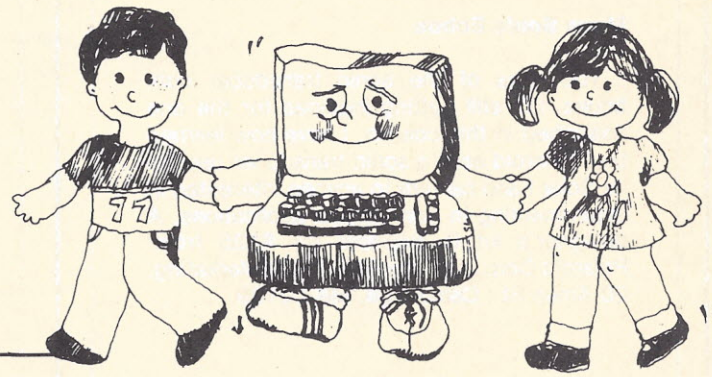
Take your scale and hold it next to the first box on the reference photo (figure 5). Line up the box on the grey scale that most nearly matches the greyness of the box in the photo. The number of the grey scale box is the digitized value of that photograph pixel. Write it into the space on your chart and continue digitizing pixels for the entire scan line. Then do the same for the other photograph.

You will end up with a set of two digitized scan lines, one from the reference photograph and one from the comparator photograph. Now refer to last month's column on the method for converting this information into three-dimensional information of the objects in the picture. Try each method described. Try your own, too, and let me know how you make out. Encounter any special problems? What is the optimum number of divisions on both the scan lines and grey-scale charts? Does it matter what type of film you use? If someone comes up with a really workable model and program, I will report on it in a future column. □

My TRS-80 Likes Me

When I Teach Kids How to Use It

by Bob Albrecht



This is a continuing series for parents and teachers who wish to help kids learn how to use, program and enjoy the TRS-80. We continue with Wandering Star as she roams outer space searching for cosmic dust.

Wandering Star learns to peek

Perhaps you have noticed that sometimes Wandering Star moves next to a cosmic dust mote, then moves away. Let's teach her how 'peek' at nearby places to see if there is any cosmic dust there. If there is, she moves directly to it instead of wandering randomly—a much more efficient way to gather food.

Recall that 'print' positions on the screen are numbered from 0 (upper left corner) to 1023 (lower right corner). For each 'print' position on the screen, there is a corresponding location in the computer's memory. These memory locations are numbered from 15360 to 16383.

- Memory location 15360 corresponds to screen position 0.
- Memory location 15361 corresponds to screen position 1.
- And so on. Memory location 16383 corresponds to screen position 1023.

You can easily compute the memory location that corresponds to a given screen position.

Memory location = screen position + 15360

You can also easily compute the screen position that corresponds to a given memory location.

Screen position = memory location - 15360

When the computer puts a character on the screen in a 'print' position, it also puts a numeric code for that character in the corresponding memory location.

The code for Wandering Star (*) is 42.
The code for cosmic dust (.) is 46.
The code for empty space (" ") is 32.

space between quotes

Try this program

```
1Ø CLS
2Ø PRINT @Ø, "*" ;
3Ø PRINT PEEK(1536Ø)
```

Here is what happened when we ran the program.

```
The asterisk is in PRINT position 0.

* 42          <-- The code was printed here by line 30 in the program.
READY
>_
```

The statement:

20 PRINT @0, "*";

tells the computer to print an asterisk (*) at print position 0. The computer also puts the code (42) for the asterisk in memory location 15360.

The statement:

30 PRINT PEEK(15360)

tells the computer to 'peek' at memory location 15360 and print what it sees there. In this case, 'peek(15360)' is 42, the code for an asterisk. The code was put in memory location 15360 by line 20.

To find out more about codes for characters, try this program.

```
1ØØ REM ** FINDING CODES FOR CHARACTERS
11Ø CLS
12Ø INPUT "WHAT CHARACTER" ; CH$
13Ø INPUT "WHERE SHOULD I PRINT IT" ; SP
14Ø CLS
15Ø PRINT @SP, CH$;
16Ø PRINT PEEK(1536Ø + SP)
```

Run it. When the computer asks 'what character?,' enter any keyboard character and press 'enter.' You can even enter a space, providing you enclose it in quotation marks (" ").

When the computer asks 'where should I print it?,' enter any whole number from 0 to 1023. The computer will then print your character where you want it and also print the code for your character by peeking at the memory location which corresponds to screen position SP.

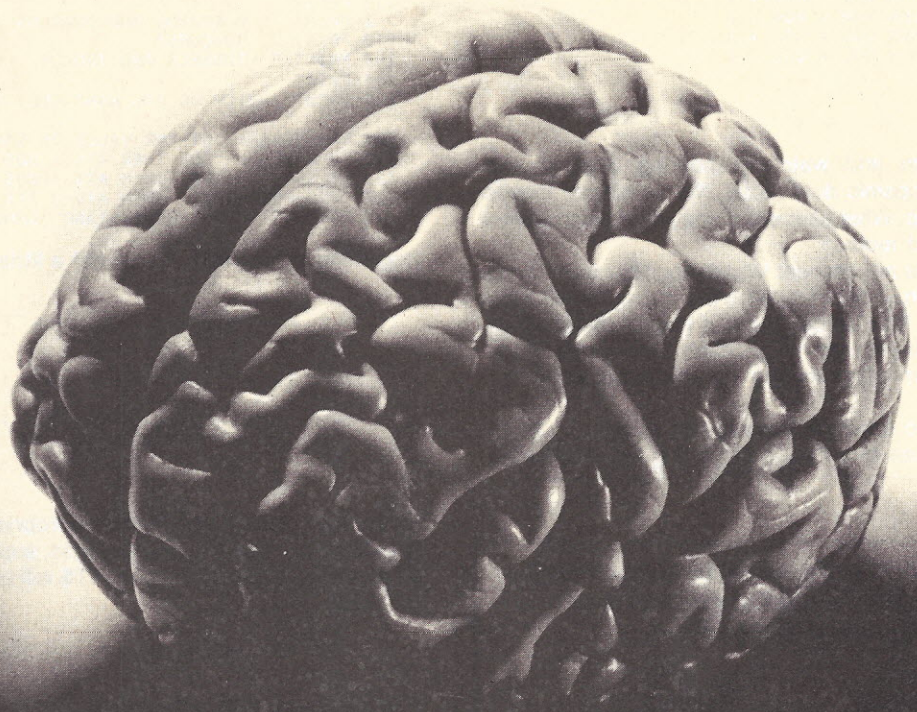
```
15Ø PRINT PEEK(1536Ø + SP )
                 ^^^^^^^^^
          This is the memory
          location which corresponds
          to screen position SP.
```

If you enter a number less than zero or more than 1023, you will, of course, get an 'error' message. So, please cooperate. Enter a whole number in the range 0 to 1023.

Here finally, is our program to teach Wandering Star how to peek at nearby places. She can peek into an adjacent screen position up, down, right or left.

```
1ØØ REM**WANDERING STAR #4
1ØØ CLS

2ØØ REM**COSMIC DUST
21Ø FOR K = 1 TO 2ØØ
22Ø   PRINT @ RND(1Ø22), ".";
23Ø NEXT K

3ØØ REM**WANDERING STAR APPEARS
31Ø ROW = 7
32Ø COL = 32
33Ø PRINT @(64*ROW + COL), "*";

4ØØ REM**WANDERING STAR RESTS
41Ø T = 2ØØØ
42Ø FOR Z = 1 TO T : NEXT Z
```

THE ULTIMATE INFORMATION MANAGEMENT SYSTEM



The brain is the perfect information management system.

Like the brain, we at Micro-Ap specialize in the management of data.

Our software is the state of the art and is designed to efficiently store and instantly report the information you need for your business and personal needs.

From inventory control to mailing list management, Micro-Ap provides the most cost effective software available.

At the heart of our systems are Micro-Ap's unique indexing and reporting methods. You are not limited to single key retrieval. Information can be referenced by zip code, date, name, or any other indices required. Operation is "menu driven" and uses screen displays with all the instructions and error sensing that allow the novice to quickly learn the system and accomplish a multitude of tasks.

Micro-Ap's Newest Industry Innovation is the SELECTOR IV™ System, with the ability to make data file conversions, arithmetic computations, global search and replace, and full page report formatting.

This floppy and hard disc oriented system is upward compatible with SELECTOR III-C2™. It adds several dimensions to the world of information management.

Experience – SELECTOR™ has been around, and improving, longer than any other data base system in microcomputers.

See the innovative SELECTOR IV™ data base manager and the standard setting GLector, general ledger system at YOUR LOCAL COMPUTER STORE or contact:

Micro-Ap
9807 Davona Dr.
San Ramon, Ca. 94583
(415) 828-6697

Lifeboat Associates
2248 Broadway
NY, NY 10024
(212) 580-0082
telex 220501

MICRO•AP

The Standard In Information Management Systems

CIRCLE INQUIRY NO. 37

```
5ØØ REM**SHE LOOKS DOWN, UP, RIGHT AND LEFT
51Ø W = Ø
52Ø DN = 1536Ø + 64*(ROW + 1) + COL
53Ø IF PEEK(DN) = 46 THEN W = 1
54Ø UP = 1536Ø + 64*(ROW - 1) + COL
55Ø IF PEEK(UP) = 46 THEN W = 2
56Ø RT = 1536Ø + 64*ROW + COL + 1
57Ø IF PEEK(RT) = 46 THEN W = 3
58Ø LT = 1536Ø + 64*ROW + COL - 1
59Ø IF PEEK(LT) = 46 THEN W = 4

6ØØ REM**SHE MOVES TO FOOD (W<>Ø) OR RANDOMLY (W=Ø)
61Ø PRINT @(64*ROW + COL), " " ;
62Ø IF W = Ø GOSUB 83Ø ELSE GOSUB 84Ø

7ØØ REM**SHOW HER IN HER NEW PLACE
71Ø PRINT @(64*ROW + COL), "*";
72Ø T = 1ØØ
73Ø FOR Z = 1 TO T : NEXT Z
74Ø GOTO 51Ø

8ØØ REM**SUBROUTINE: TWO ENTRY POINTS
81Ø REM**ENTER AT LINE 83Ø IF WS MOVES AT RANDOM
82Ø REM**ENTER AT LINE 84Ø IF SHE HAS SENSED FOOD
83Ø W = RND(4)
84Ø IF W = 1 THEN ROW = ROW + 1
85Ø IF W = 2 THEN ROW = ROW - 1
86Ø IF W = 3 THEN COL = COL + 1
87Ø IF W = 4 THEN COL = COL - 1
88Ø RETURN

999 END
```

Run this program and watch Wandering Star. If she wanders next to a cosmic dust mote, she senses it, moves right in and zaps it up. What happens if she moves next to two or more cosmic dust motes?

This program does not prevent Wandering Star from trying to wander off screen. If this happens, the computer will stop with an '?FC error.' You, of course, can modify the program so that she stays on-screen or, in case she wanders off-screen, might return after finding no cosmic dust in the cosmic desert.

Do you want us to continue the saga of Wandering Star? Would you like to see other ways of writing the Wandering Star programs we have done so far? Would you like particular things described in more detail? What would you like? Write us in care of Interface Age. Parts 1 to 3 of this series are available free in an 8-page booklet from Sharon Ross, Radio Shack, Dept. 3, 1300 One Tandy Center, Ft. Worth, TX 76102.

Gamemaster's dice

Have you figured out how our program to create an adventurer works? We assume that you had little or no trouble with lines 100 through 340, shown below.

```
1ØØ REM**CREATE A FANTASY CHARACTER FOR
12Ø REM**D & D, RUNEQUEST OR T & T
13Ø REM**SET UP ATTRIBUTE STRINGS
14Ø DD$ = "  6 STR INT WIS CON DEX CHA "
15Ø RQ$ = "  7 STR INT POW CON DEX CHA SIZ "
16Ø TT$ = "  6 STR IQ  LK  CON DEX CHR "

2ØØ REM**TELL ABOUT THE PROGRAM
21Ø CLS
22Ø PRINT "I CAN CREATE A CHARACTER FOR"
23Ø PRINT
24Ø PRINT "  DUNGEONS AND DRAGONS (DD)"
25Ø PRINT "  RUNQUEST             (RQ)"
26Ø PRINT "  TUNNELS AND TROLLS   (TT)"
27Ø PRINT
28Ø INPUT "WHICH DO YOU WANT (DD, RQ, OR TT)"; GAME$

3ØØ REM**CHECK OUT THE VALUE OF GAME$
31Ø IF GAME$ = "DD" THEN AT$ = DD$ : GOTO 41Ø
32Ø IF GAME$ = "RQ" THEN AT$ = RQ$ : GOTO 41Ø
33Ø IF GAME$ = "TT" THEN AT$ = TT$ : GOTO 41Ø
34Ø PRINT "I DON'T UNDERSTAND" GAME$ : GOTO 27Ø
```

Beginning at line 400, things get a little stickier.

```
4ØØ REM**ROLL THE CHARACTER
41Ø NA$ = LEFT$(AT$, 4)
42Ø NA = VAL(NA$)
43Ø FOR K = 1 TO NA
44Ø   AT = 4*K + 1
45Ø   DICE = RND(6) + RND(6) + RND(6)
46Ø   PRINT MID$(AT$, AT, 4), DICE
47Ø NEXT K
```

Remember, AT$ is set to the appropriate attribute string in lines 310 through 330. For example, suppose we enter "RQ" in response to line 280. Then, AT$ will be set equal to RQ$ in line 320.

```
AT$ = "  7 STR INT POW CON DEX CHA SIZ "
```

In this case, NA$ will become " 7" in line 410. Thus, NA becomes 7 in line 420. The 'for-next' loop in lines 430 through 470 will be done for K = 1 to 7.

K = 1

```
AT = 4*K + 1 = 4*1 = 5
DICE will be a random number from 3 to 18
MID$(AT$, AT, 4) = MID$(AT$, 5, 4)
                 = "STR "
```

So, the computer will print the string "str " followed by the value of DICE.

K = 2

```
AT = 4*K + 1 = 4*2 + 1 = 9
DICE will be a random number from 3 to 18.
MID$(AT$, AT, 4) = MID$(AT$, 9, 4)
                 = "INT "
```

So, the computer will print the string "int " followed by the value of DICE. And so on up to K = 7.

We assume that you understand the rest of the program shown below. True?

```
5ØØ REM**ASK IF SOMEONE WANTS ANOTHER CHARACTER
51Ø PRINT
52Ø PRINT "FOR ANOTHER CHARACTER, PRESS THE SPACE BAR";
53Ø K$ = INKEY$ : IF K$ = "" THEN 53Ø
54Ø IF K$ = " " THEN 21Ø ELSE 53Ø

999 END
```

Programming problems

Did you try our problems last issue? Here are some solutions to Problem #1 positive, negative or zero.

We expect many of you thought of this solution.

```
100 REM:::PROBLEM #1 POSITIVE, NEGATIVE OR ZERO
110 REM:::RECREATIONAL COMPUTING, JAN/FEB 1980
120 CLS

300 REM:::ASK FOR A NUMBER, X
310 PRINT : INPUT "NUMBER, PLEASE" ; X

500 REM:::TELL WHETHER NUMBER IS POSITIVE,
    NEGATIVE OR ZERO
510 ON SGN(X) + 2 GOTO 520, 530, 540
520 PRINT "YOUR NUMBER IS NEGATIVE" : GOTO 310
530 PRINT "YOUR NUMBER IS ZERO"     : GOTO 310
540 PRINT "YOUR NUMBER IS POSITIVE" : GOTO 310

999 END
```

How would you use 'on. . .gosub' instead of 'on. . .goto?' We bet that not many of you did it this way.

```
100 REM:::PROBLEM #1 POSITIVE, NEGATIVE OR ZERO
110 REM:::RECREATIONAL COMPUTING, JAN/FEB 1980
120 CLS

200 REM:::DEFINE STRING ARRAY OF POSSIBILITIES
210 A$(1) = "NEGATIVE"
220 A$(2) = "ZERO"
230 A$(3) = "POSITIVE"

300 REM:::ASK FOR A NUMBER, X
310 PRINT : INPUT "NUMBER, PLEASE" ; X

510 REM:::TELL WHETHER NUMBER IS POSITIVE,
    NEGATIVE OR ZERO
520 WHICH = SGN(X) + 2
530 PRINT "YOUR NUMBER IS " A$(WHICH)

700 REM:::GOTO 'ASK FOR A NUMBER, X'
710 GOTO 310

999 END
```

There are still many more ways to do this problem. Suppose:

```
A$ = "NEGATIVEZERO    POSITIVE"
       8 places 8 places 8 places

or   A$ = "NEGATIVE, ZERO, POSITIVE"
```

In the above cases, how would you write the program segment beginning at line 510?

Shall we continue our Programming Problems section?

Computertown, USA!

Now happening:

Computerstore, USA is a low-profile "store" that stocks and sells to kids and others, especially good software for the TRS-80 and the Commodore Pet.

Computerkid, USA is a course to help kids learn to use, program and enjoy computers. In most courses, people come to the computers on some regular schedule; in this course, the computer comes to the people.

It goes like this: You sign up for the course and attend the first meeting in which you learn some essentials such as how to turn on the computer. Then, you take the computer home for two weeks, along with a "teach yourself" book and other written materials.

During the two weeks, if you have questions, call the Computerkid, USA Hotline. Kids will answer your questions. At the end of the first week, you can attend the regular Computerkid, USA meeting to share your experiences with other learners and the "instructors." At the end of the second week, you attend the meeting, check in your computer and sign up for the second course.

We expect that Computerkid, USA will be a family affair. The child enrolls in the course, takes a computer home and the entire family learns to use, program and enjoy it.

For info send a stamped, self-addressed envelope to Computertown, USA, P.O. Box 310, Menlo Park, CA 94025. □

Learning with Micros

By Louis E. Frenzel, Jr.

Learning to Write CAI Programs

Many educators shy away from using computer aided instruction (CAI) due to the lack of courseware. Some teaching software does exist, of course, but not enough to plunge teachers into investing in microcomputers. An immediate solution would be to write your own, but fear of the unknown or failure prevents most people from trying. While CAI development is not simple, neither is it very difficult. Like writing, drawing or sewing, it is possible to learn.

Developing learning materials

The approach should be to consider CAI as a learning supplement. Perhaps to complement a lecture, provide remedial work, give drill and practice, or present a quiz. In any case, CAI is part of a much broader course.

Here is a step-by-step procedure:

1. *List learning objectives.* Try to express them in behavioral terms, that is, terms that identify measurable accomplishments. Be very specific about what knowledge or skills the student must demonstrate.
2. *Develop the program outline.* Try to make it as detailed as possible. The more detailed, the easier it will be to sequence and write material later. Try to modularize the program by grouping it into major sections, so each section is a separate program covering one topic. These modules can then be sequentially linked into a total teaching package.
3. *Write instructional materials on paper.* A tutorial should be in programmed instruction (PI) format, that is, break up the material into short facts and concepts called frames, each frame containing an idea or item of information flashed on a CRT screen, followed by a question that reinforces the information and results in interaction with the system. Use multiple choice, true/false or single-word or numeral fill-ins.

The simplest PI uses the linear format where one frame simply follows the other. After each question, the answer is verified and the next frame of information is presented. A more complex format called branching PI uses multiple choice questions that lead to multiple teaching frames. If the correct answer is given, the response is recognized and the next frame of information begins. If an incorrect answer is given, the program branches to a frame of remedial information or a hint before the next regular frame of information is presented.

4. *Develop module examinations.* Generate a multiple question quiz. It is best to stop after each module and review with an examination. Present the questions to the student one at a time on the screen and have the program keep track of the right and wrong answers. Once the quiz is over, the student's score can be tallied.
5. *Edit the material.* Keep in mind that the only material in the program is that which accomplishes the originally-stated learning objective. Keep the material lean so that it can be learned quickly, and minimize computer memory space.
6. *Write the program.* It is best to do this on a module-by-module basis in order to test each section as you go along.
7. *Debug the program.* Once written, you will need to execute it and make whatever changes and additions necessary.
8. *Test and validate the program.* Try it out on someone who does not know the material. See if learning actually does take place. You must get feedback as to whether the program works and learning occurs.
9. *Use the program.* Test it out on an entire class. This will give feedback as to its effectiveness. With practical use, you will find ways to improve the program and correct errors. If this is your first programming job, it will probably be desirable for you to concentrate on the written material and minimize the use of graphics.

Trying to use the computer to draw pictures and do other fancy graphical techniques is fun, interesting and valuable. But it is also difficult and time consuming for the novice writer. If figures, illustrations, or other pictorials are required, hand draw them on extra sheets of paper and hand them out. Most students won't object, and it will save you a lot of time and effort in the beginning. Include supplementary written material if it would be helpful. This makes the program more multi-media rather than just pure CAI.

CAI should be considered a learning supplement to give drill and practice, or present a quiz.

Another hint is keep each frame of information short and to the point. This won't be too difficult since you will be working with a limited amount of CRT screen space. For example, 24 lines of 40 characters each is typical of many microcomputers. Some of the larger and more sophisticated machines feature 24-80 character lines. Don't write frames of information requiring the display of several sequential screens of information. Using a lot of text requires a lot of reading which may slow down the student. It also uses up expensive memory space quickly.

Keep the entire teaching module short, say less than 15 minutes. It is difficult to hold someone's attention for a long period of time. By keeping the frames as well as each module short, you will be able to keep the attention of most students. Work in a lot of key pushing and references to external materials so that the program gives the feeling of being interactive and fast paced.

Language preferences

One of the easiest ways to write a teaching program is to use Pilot, a special computer language designed specifically for writing CAI. It is extremely easy to use, even simpler than Basic. Pilot is available for most popular microcomputers including the Apple II, Bell and Howell's version of the Apple, Commodore PET, Heath/Zenith H-89/Z-89 and the SWTP 6800. Check with the manufacturer of your machine to verify its availability. It is a highly recommended way to ease and speed up the development of CAI learning programs.

Some microcomputer manufacturers offer special CAI development software. These are unique operating systems or languages that help individuals create their own learning programs without knowing how to program. These packages come with complete documentation and self-directing software that greatly simplifies the CAI process. Typical is the Bell & Howell (Apple II) Genis I system, which contains Pilot. □

COMPUTER BOOKS THAT TALK YOUR LANGUAGE . . . FROM BEGINNER TO EXPERT.

Howard W. Sams is the source for the best and latest in computer technology. The world's leading authors. Clear, concise, easy-to-understand text. Fully illustrated for better comprehension. Actual programs for today,s personal computers. Informative and entertaining. Titles for everyone . . . from the novice to the hobbyist to the expert.

THE STARTERS

Entry-level books for the beginner can bring the novice up to a high level of knowledge quickly.

☐ **THE HOWARD W. SAMS CRASH COURSE IN MICROCOMPUTERS**
The short-cut to understand microcomputers. Spend a few hours with this self-instructional course and gain a working knowledge of all aspects of microcomputers from fundamentals to operating systems, programming, peripheral equipment and much, much more. 264 pages. No. 21634. By Frenzel. $17.50.

☐ **MICROCOMPUTERS FOR BUSINESS APPLICATIONS**
A microcomputer expert discusses the benefits of microcomputer systems for small business owners. Explains types of microcomputers available, points out pitfalls to avoid, and defines computer-related terms in easy-to-understand language. 256 pages. By Barden. No. 21583. $8.95.

☐ **A GUIDEBOOK TO SMALL COMPUTERS**
If you are contemplating buying a small computer system for home or business, this book can save you time and trouble. Covers fundamentals of small computers including hardware and software. Includes information on manufacturers' warranties, service and a handy directory of small computer manufacturers. By Barden. No. 21698. $5.95.

☐ **GETTING ACQUAINTED WITH MICROCOMPUTERS**
Provides a complete working knowledge of microcomputer organization, operation, and programming. 288 pages. By Frenzel. No. 21486. $8.95

☐ **FUNDAMENTALS OF DIGITAL COMPUTERS (2nd ED.)**
Unravels the mysteries of computers and programming. 320 pages. By Spencer. No. 21534. $9.95.

☐ **INTRODUCTION TO MICROCOMPUTERS FOR THE HAM SHACK**
Gives the radio amateur an opportunity to be in the forefront of utilizing and developing techniques in "computercations." 96 pages. By Helms. No. 21681. $4.95.

LEARN ABOUT MICROCOMPUTERS, SAVE 10%...AND GET A FREE GIFT!

☐ **THE COMPUTER PRIMER LIBRARY**
This 3-volume "starter" library was selected for the novice who wants to learn about microcomputers. Concepts. Programming. Computer graphics. Languages. Games. By Mitchell Waite and Michael Pardee—two of the leading authors in the field of computers.

MICROCOMPUTER PRIMER—368 pages of up-to-date facts on microcomputers and computer concept.

COMPUTER GRAPHICS PRIMER—Over 20 programs for creating your own video graphics (complex drawings, plans, maps, and schematics) on the new 6502-based personal computers.

BASIC PROGRAMMING PRIMER—Provides a working knowledge of BASIC, the most popular computer language today. Explains fundamentals, of BASIC, program control, organization, additional functions, variations, and includes a game program.

3-VOLUME LIBRARY PACKAGE ONLY $30.47—includes a FREE COPY of YOUR OWN COMPUTER—an introduction to home computers. No. 21706

THE PROGRAMMERS

A computer is only as good as the software. Sams offers several books that explain programming and tell you how to develop, debug and test programs for your equipment.

☐ **PROGRAMMING & INTERFACING THE 6502, WITH EXPERIMENTS**
Dr. De Jong has compiled 14 chapters of information and programs for 6502-based microcomputer systems. Experiments and examples are written so that a KIM, AIM or SYM system may be used to reinforce the material in each chapter. 448 pages. By De Jong. No. 21651. $11.95.

☐ **HOW TO PROGRAM AND INTERFACE THE 6800**
Discusses the internal structure, instruction set, and programming techniques for the 6800. Experiments demonstrate "real world" applications. By Staugaard. 432 pages. No. 21684. $13.95.

☐ **DBUG: AN 8080 INTERPRETIVE DEBUGGER**
Covers program operation and how it's applied to program development and testing. 112 pages. By Titus and Titus. No. 21536. $4.95.

☐ **6502 SOFTWARE DESIGN**
Tells how to program for the 6502 assembly language. 288 pages. By Scanlon. No. 21656. $10.50.

☐ **8080/8085 SOFTWARE DESIGN—2 VOLUMES**

☐ VOLUME 1 gives you an introduction to assembly language programming. 336 pages. By Titus, Larsen, Rony, & Titus. No. 21541. $9.50

☐ VOLUME 2 is a unique, one-of-a-kind computer science book for the design engineer. Written in Intel machine code. 352 pages. By Titus, Larsen, & Titus. No. 21615. $9.95.

☐ **TWO-VOLUME SET No. 21659 ONLY $17.50**

☐ **TEA: AN 8080/8085 CO-RESIDENT EDITOR-ASSEMBLER.**
256 pages. No. 21628. By Titus. $8.95.

THE COMPUTER TECHNOLOGY LEADERS

Even if you're an avid computer buff, rapidly changing technology makes it difficult to stay current. Sams has a complete line of books and reference publications to help you keep abreast of the current state-of-the-art.

☐ **MICROCOMPUTER-ANALOG CONVERTER SOFTWARE & HARDWARE INTERFACING.**
Concepts and techniques of interfacing digital computers to analog devices. 288 pages. By Titus, Titus, Rony, & Larsen. No. 21540. $9.50.

☐ **TRS-80 INTERFACING.**
192 pages. No. 21633. By Titus. $8.95.

☐ **INTERFACING & SCIENTIFIC DATA COMMUNICATIONS EXPERIMENTS.**
160 pages. By Rony, Larsen, Titus, &Titus. No. 21546. $5.95.

☐ **MICROCOMPUTER INTERFACING WITH THE 8255 PPI CHIP.**
224 pages. By Goldsbrough. No. 21614. $8.95.

☐ **Z-80 MICROPROCESSOR PROGRAMMING & INTERFACING—2 VOLUMES.**
800 pages. By Nichols, Nichols, & Rony. No. 21611. $21.95.

☐ **INTRODUCTORY EXPERIMENTS IN DIGITAL ELECTRONICS AND 8080A MICROCOMPUTER PROGRAMMING AND INTERFACING—2 VOLUMES.**
912 pages. By Rony, Larsen, & Titus. No. 21552. $20.95.

A BEST-SELLER . . . A TREMENDOUS VALUE!

This is the most comprehensive reference available on all phases of computers and their applications. Contains over 22,000 definitions, acronyms, and abbreviations. 14 appendices. Hardbound. 944 pages. By Sippl & Sippl. No. 21632. $29.95.

☐ **COMPUTER DICTIONARY & HANDBOOK (3RD EDITION)**

THE LOGIC COOKBOOKS

☐ **LANCASTER'S COOKBOOK LIBRARY**
A famous resource for all electonics buffs who want to know what makes TTLs, CMOS, and active filters cook—what they are, how they work and how to use them. By Don Lancaster, one of the most popular authors in the electronics industry.

TTL Cookbook No. 21035	$ 9.50
CMOS Cookbook No. 21398	$10.50
Active-Filter Cookbook No. 21168	$14.95
TOTAL LIST PRICE	$34.95
Less 15% Discount	5.24
YOU PAY ONLY	$29.71

☐ **ORDER SPECIAL LIBRARY PACKAGE NO. 21707 AND SAVE!**

CIRCLE INQUIRY NO. 55

ORDER FORM

HOWARD W. SAMS & CO., INC.
4300 West 62nd Street, P.O. Box 7092
Indianapolis, Indiana 46206
(317) 298-5400

Indicate quantity in boxes above and complete ordering information below.

Sub Total_____

Add local sales tax where applicable_____

GRAND TOTAL_____

☐ PAYMENT ENCLOSED
☐ CHECK ☐ MONEY ORDER
☐ MASTER CHARGE ☐ BANK AMERICARD/VISA

Exp. Date_____

Account No._____

Interbank No. (Master Charge only)_____
Minimum Credit card purchase $10 AD014

Name (print)_____

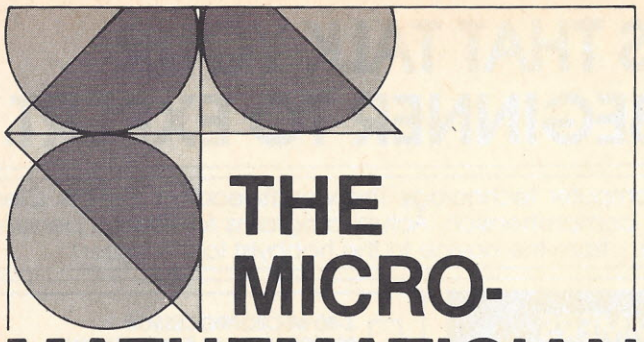
Signature_____

Address_____

City_____ State_____ Zip_____

Prices subject to change without notice. All books available from Sams Distributors, Bookstores, and Computer Stores. Offer good in U.S. only. In Canada, contact Lenbrook Industries Ltd., Scarborough, M1H 1H5, Ontario, Canada.

Howard W. Sams & Co., Inc.



THE MICRO-MATHEMATICIAN

by Dr. John C. Nash

Generalized Inverse Matrices: A Practical Tool for Matrix Methods on Microcomputers

Matrix methods pervade the practice of computation in all the sciences—physical, social and biological. Following is an extension of the discussion of matrix calculus along with a powerful program that enables a variety of matrix computations.

Definitions

To begin, take the matrix A. If A is square (m = n, where A has m rows and n columns) and non-singular (that is, its determinant is non-zero or alternatively that no row/column is equal to a weighted sum of all the rest of the rows/columns), then the *inverse* of A, denoted A^{-1}, can be calculated so that

$$A\,A^{-1} = A^{-1}\,A = I$$

where I is the identity matrix (1's on the diagonal, zeros elsewhere). If A is of order n, so are I and A^{-1}. When A is not square ($m \neq n$) or is singular, the inverse is no longer defined. However, a variety of generalized inverses can be proposed. Perhaps the most useful of these is the Moore-Penrose generalized inverse A^+ which is required to have the following properties:

1) $A\,A^+\,A = A$
2) $A^+\,A\,A^+ = A^+$
3) $(A^+\,A)^T = A^+\,A$
4) $(A\,A^+)^T = A\,A^+$

where the ^{T} denotes matrix transposition—the interchange of rows and columns. Other generalized inverses can be defined which only satisfy some of these conditions.

Applications

The Moore-Penrose inverse is extremely versatile and has a large literature. For instance, there is a 270-page annotated bibliography containing references to 1776 works in the book by M.Z. Nashed, *Generalized Inverses and Applications* (Academic Press, New York 1976). Here a few major tasks will be outlined.

a) Least squares solutions.

Given m values y_i, $i = 1,2,\ldots,m$, and a data matrix A_{ij}, $i = 1,2,\ldots,m$, $j = 1,2,\ldots,n$, a least squares solution

$$\underline{x} = (x_1, x_2, x_3, \ldots, x_n)^T$$

minimizes the sum of squares

$$S = \sum_{i=1}^{m} (y_i - \sum_{j=1}^{n} A_{ij}x_j)^2$$

This is the foundation of curve fitting and a great part of the study of statistics. In matrix notation, we write the *residual* vector

$$\underline{r} = \underline{y} - A\,\underline{x}$$

and express the sum of squares as

$$S = \underline{r}^T\,\underline{r}$$

The use of calculus to minimize S gives the *normal equations*

$$A^T\,A\,\underline{x} = A^T\,\underline{y}$$

so that if $(A^T\,A)$ is non-singular, we take its inverse and multiply through to get

$$\underline{x} = (A^T\,A)^{-1}\,A^T\,\underline{y}$$

Using the properties of the Moore-Penrose inverse, it is quite easy to show that

$$\underline{x}^* = A^+\,\underline{y}$$

also minimizes S. Moreover, if $(A^T\,A)$ is singular, there are infinitely many least squares solutions, but of these x^* is the one having *minimum length,* that is, which minimizes $x^T\,x$ at the same time as it minimizes S. It turns out that x^* is unique but that *all* solutions to the least squares problem can be written in the form

$$\underline{x} = \underline{x}^* + (I_n - A^+\,A)\,\underline{e} = A^+\,\underline{y} + (I_n - A^+\,A)\,\underline{e}$$

where e is an *arbitrary* vector (any numbers will do) and I_n is the identity matrix of order n.

b) Linear equations.

If m = n above, then it is possible that

$$\underline{y} = A\,\underline{x}$$

so that we have a set of n simultaneous linear equations in n unknowns x. By computing

$$\underline{x}^* = A^+\,\underline{y}$$

we obtain a least squares solution, and if the minimum sum of squares is zero, we have a solution to the linear equation problem. If the sum of squares S is not zero, our equations are inconsistent. Note that a zero sum of squares only means we have *a* solution—not *the* solution unless A is non-singular.

c) Inverse of square matrices.

When A is square (n by n) and non-singular,

$$A^+ = A^{-1}.$$

From the general expression for any least squares solution

$$\underline{x} = A^+\,\underline{y} + (I - A^+\,A)\,\underline{e}$$

it is easily seen that linear equations in which A is non-singular are uniquely solved by

$$\underline{x} = A^{-1}\,\underline{y}$$

because in such cases

$$(I - A^+\,A) = 0,$$

the null matrix. This could provide a test of whether or not A^+ was really equal to A^{-1}, hence of the singularity of A.

Methods

While some direct methods exist for computing generalized inverses, the calculations are generally performed by breaking them down to simpler parts. Without going into details, a particularly powerful technique is to perform the singular value decomposition (svd)

$$A = U\,Z\,V^T$$

where A (m by n) is decomposed into U (m by n) with

$$U^T\,U = I_n$$

and V (n by n) with

$$V^T\,V = V\,V^T = I_n$$

and Z (n by n) diagonal, that is, only the diagonal elements are non-zero. In the decomposition, the elements of Z are required to be non-negative. The svd can be computed in a number of ways. On large machines, one well-known and proven algorithm is that of Golub and Reinsch. This has been

CP/M* compatible software

SYSTEM MAINTENANCE

DIAGNOSTICS I: Easily the most comprehensive set of CP/M compatible system check-out programs ever assembled. Finds hardware errors in your system, confirms suspicions, or just gives your system a clean bill of health. Tests:

- Memory
- CPU (8080/8085/Z80)
- Terminal
- Disk
- Printer

To our knowledge the CPU test is the first of its kind anywhere. Diagnostics I can help you find problems before they become serious. A good set of diagnostic routines are a must in any program library.

Minimal requirements: 24K CP/M. Supplied with complete user manual: $60.00 Manual alone: $15.00.

ACCOUNTING

ACCOUNTS PAYABLE/RECEIVABLE: A complete, user oriented package which features:

automatic postings to general ledger (optional)

accounts payable:
- check printing with invoice
- invoice aging

accounts receivable:
- progress billing
- customer statements
- partial invoice payments
- invoice aging

The entire package is menu driven and easy to learn and use. It incorporates error checking and excellent user displays. This package can be used stand alone or with the General Ledger below.

Supplied with extensive user manual: $200.00. Manual alone: $20.00.

GENERAL LEDGER: A complete, user oriented package which features:

- Accepts postings from external programs (i.e. AP/AR above)
- Accepts directly entered postings
- Maintains account balances for current month, quarter, and year and previous three quarters
- Financial reports: trial balance, income statement balance sheet, and more.

Completely menu driven and easy to learn and use. Excellent displays and error checking for trouble free operation. Can be used stand alone or with Accounts Payable/Receivable above.

Supplied with extensive user manual: $200.00. Manual alone: $20.00.

Both require 48K CP/M, terminal with cursor positioning, home and clear home, one 8" disk or Two 5" disks. CBASIC2 required.

TEXT PROCESSING

TFS—Text Formatting System: An extremely powerful formatter. More than 50 commands. **Supports all major features including:**

- left & right margin justification
- user defined macros
- dynamic insertion from disk file
- underlining and backspace

TFS lets you make multiple copies of any text. For example: Personalized form letters complete with name & address & other insertions from a disk file. Text is not limited to the size of RAM making TFS perfect for reports or any big job.

Text is entered using CP/M standard editor or most any CP/M compatible editor. TFS will link completely with Super-M-List making personalized form letters easy.

Requires: 24K CP/M.

Supplied with extensive user manual: $85.00. Manual alone: $20.00.

Source to TFS in 8080 assembler (can be assembled using standard CP/M assembler) plus user manual: $250.00.

MAILING LIST

SUPER-M-LIST: A complete, easy to use mailing list program package. Allows for two names, two address, city, state, zip and a three digit code field for added flexibility. Super-M-List can sort on any field and produce mailing labels direct to printer or disk file for later printing or use by other programs. Super-M-List is the perfect companion to TFS. Handles 1981 Zip Codes!

Requires: 48K CP/M.

Supplied with complete user manual: $75.00. Manual alone: $10.00

UTILITIES

Utility pack #1: A collection of programs that you will find useful and maybe even necessary in your daily work (we did!). Includes:

CMP:	Compare two files for equality.
ARCHIVER:	Compacts many files into one, useful when you run out of directory entries.
SORT:	In core sort of variable length records.
XDIR:	Extended, alphabetical directory listing with groupings by common extension.
PRINT:	Formatted listings to printer.
PG:	Lists files to CRT a page at a time.

. . . plus more . . .

Requires: 24K CP/M.

Supplied with instructions on discette: $50.00.

PROGRAMMING LANGUAGES

FORTH: a full, extended FORTH interpreter/compiler produces COMPACT, ROMABLE code. As fast as compiled FORTRAN, as easy to use as interactive BASIC.

NEW!

SELF COMPILING: Includes every line of source code necessary to recompile itself.
EXTENSIBLE: Adds functions at will.
Z80 & 8080 ASSEMBLERS included
Single license, OEM licensing available
Please specify CPU type: Z80 or 8080
Supplied with extensive user manual and tutorial: $150.00
Documentation alone: $25.00

ENHANCED 'TINY' PASCAL: We still call it 'Tiny' but it's bigger and better than ever! This is the Famous Chung/Yuen 'Tiny' Pascal with more features added. Features include:

- recursive procedures/functions
- integer arithmetic
- CASE
- FOR (loop)
- sequential disk I/O
- one dimensional arrays
- IF . . . THEN . . . ELSE
- WHILE
- 'PEAK' & 'POKE'
- READ & WRITE
- REPEAT . . . UNTIL
- more

'Tiny' Pascal is fast. Programs execute up to ten times faster than similar BASIC programs.

SOURCE TOO! We still distribute source, in 'Tiny' Pascal, on each discette sold. You can even recompile the compiler, add features or just gain insight into compiler construction.

'Tiny' Pascal is perfect for writing text processors, real time control systems, virtually any application which requires high speed. Requires: 36K CP/M. Supplied with complete user manual and source on discette: $85.00.

Manual alone: $10.00.

SOFTWARE SECURITY

ENCODE/DECODE: A complete software security system for CP/M. Encode/Decode is a sophisticated coding program package which transforms data stored on disk into coded text which is completely unrecognizable. Encode/Decode supports multiple security levels and passwords. A user defined combination (One billion possible) is used to code and decode a file. Uses are unlimited. Below are a few examples:

- data bases
- general ledger
- inventory
- payroll files
- correspondence
- accounts pay/rec
- programs
- tax records
- mailing lists

Encode/Decode is available in two versions:

Encode/Decode I provides a level of security suitable for normal use. Encode/Decode II provides enhanced security for the most demanding needs. Both versions come supplied on discette and with a complete user manual.

Encode/Decode I: $50.00
Encode/Decode II: $100.00 Manual alone: $15.00

INTERCOMPUTER COMMUNICATIONS

TERM: a complete intercommunications package for linking your computer to other computers. Link either to other CP/M computers or to large timesharing systems. TERM is comparable to other systems but costs less, delivers more and source is provided on discette!

With TERM you can send and receive ASCII and Hex files (COM too, with included convertion program) with any other CP/M computer which has TERM or compatible package. Allows real time communication between users on separate systems as well as acting as timesharing terminal.

- Engage/disengage printer
- error checking and auto retry
- terminal mode for timesharing between systems
- conversational mode
- send files
- receive files

Requires: 32K CP/M.
Supplied with user manual and 8080 source code: $110.00
Manual alone: $15.00.

CP/M Formats: 8" soft sectored, 5" Northstar, 5" Micropolis Mod II, Vector MZ

All Orders and General Information:
SUPERSOFT ASSOCIATES
P.O. BOX 1628
CHAMPAIGN, IL 61820
(217) 359-2112
Technical Hot Line: (217) 359-2691
(answered only when technician is available)

VISA

master charge

SuperSoft
First in Software Technology

*CP/M REGISTERED TRADEMARK DIGITAL RESEARCH

translated into Basic by Kris Stewart and published at part of *SCRUNCH—Numerical computations on very small machines* (California Software Co., El Cerrito, CA, 1979). In my own work, and in the program below, I use a variant of the Jacobi algorithm of 1846 (yes 1846!) described in my book *Compact numerical methods for computers: linear algebra and function minimisation* (Wiley Halsted, New York 1979).

Having the svd of A makes it very easy to find A^+. For if we define

$$S^+_{ii} = 1/S_{ii} \quad \text{if } S_{ii} > 0$$
$$= 0 \quad \text{if } S_{ii} = 0$$

then it is easily shown that

$$A^+ = V S^+ U^T$$

satisfies all four of the Penrose conditions given earlier.

A side benefit of the svd is that if A is square, symmetric (that is $A_{ji} = A_{ij}$) and positive definite, so that

$$\underline{x}^T A \underline{x} > 0$$

for all non-null $\underline{x}$, then the values S_{ii}, i = 1,2,. . .,n, called singular values, also happen to be the eigenvalues of A. These eigenvalues correspond to the frequencies of vibration of structures—bridges, spacecraft, molecules or even economies of nations. For any symmetric matrix A, we can shift the eigenvalues to produce an appropriate positive definite matrix

$$A' = A + k I$$

so that the eigenvalues of A are $(S'_{ii}-k)$ for i = 1,2,. . .,n. The eigenvectors (normal modes of vibration) of both A and A′ are the columns of the matrix V. Closely related to this, in statistics, the matrices U and V define the principal coordinates and components of a data matrix.

When solving least squares or linear equations problems, it is worthwhile noting that it is unnecessary to compute A^+ explicitly since

$$\underline{x}^* = A^+ \underline{y} = V S^+ (U^T \underline{y}) = V (S^+ \underline{w}) = V \underline{q}$$

where $\underline{w} = U^T \underline{y}$ and $\underline{q} = S^+ \underline{w}$. In this case we save the storage of a full matrix relative to a single vector.

A subroutine and two drivers

Listing 1 (parts a and b) presents a Basic subroutine called 'geninv' (lines 500 to 1290) which performs the singular value decomposition of a matrix A, which is initially in an array of the same name. During the calculation, matrix U is computed in array A, the generalized inverse A^+ is formed in array B, matrix V in array V, Z in array (vector) Z. A tolerance for zero, T9, is needed. The choice of tolerances is a difficult subject, and the interested reader is urged to consult references on numerical methods and to experiment with various values for the tolerance in the programs given. Zero is a valid choice for T9, but the iteration used in 'geninv' may then fail to converge. A limit on the number of sweeps C8 could be used between lines 1015 and 1020. A limit of 30 sweeps is reasonable in most instances.

Several parts of the subroutine are included for generality and can be omitted to save array and program storage space for particular applications. Listing 1 also includes a subroutine to copy array A to array C (lines 1500 to 1540) and a subroutine to compute a residual vector and its sum of squares (lines 2000 to 2090).

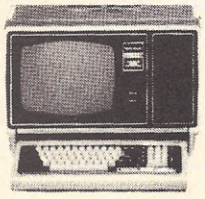
LISTING 1A

```
500 REM SUBROUTINE TO COMPUTE A+ --  GENINV -- J C NASH 1979
510 REM SOURCE\ J. C. NASH
520 REM COMPACT NUMERICAL METHODS FOR COMPUTERS
530 REM WILEY HALSTED, NEW YORK, 1979
540 REM ALGORITHM #1 AND #2
550 REM INITIALIZE V, U OVERWRITES A
560 FOR I=1 TO N
570 FOR J=1 TO N\V(I,J)=0\NEXT J
580 V(I,I)=1
590 NEXT I
```

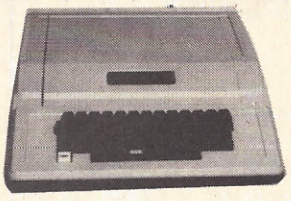
ΩMEGA SALES CO.

"WHOLESALE COMPUTER PRICES" DIRECT TO THE PUBLIC

12 Meeting St., Cumberland, R.I. 02864



TRS•80 ™
Model II - $3,500



Apple II
16K - $1049

Atari 800 - $749

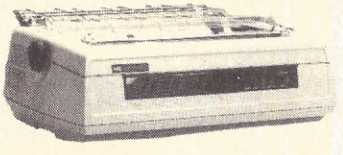
PRODUCT SPECIAL OF THE MONTH!!

INTERTEC SUPERBRAIN™
32K RAM - $2449.00
64K RAM - $2649.00

Products are NOW IN STOCK AT ΩMEGA Sales Co.

For: TRS•80, Apple, CBM,
(Interface Included)

Epson TX-80 - $745



NEC Spinwriter
5510-5530 - $2449

Soroc 120 - $699

CALL TOLL FREE FOR ΩMEGA'S PRICE!

ΩMEGA OFFERS THE BEST DELIVERY AND PRICE ON:
APPLE • ATARI • TRS·80*MODEL II • INTERTEC •
T.I. 810 • HEWLETT-PACKARD-85 • SOROC •
COMMODORE • NEC • QUME • CENTRONICS

ΩMEGA sells only factory fresh, top quality merchandise to our customers.
ΩMEGA will try to match any current advertised price with similar purchase conditions.
Before you buy anywhere else - be sure to call ΩMEGA Sales Co.
1-401-722-1027 or

CIRCLE INQUIRY NO. 48

ΩMEGA TOLL FREE 1-800-556-7587

ΩMEGA ships via UPS, truck, or air. COD's, VISA, Mastercharge accepted.
"A member in good standing of the better business bureau."

VISA master charge

TRS-80 is a trademark of Tandy Corp.

```
600 C8=0 \ REM COUNT SWEEPS
620 REM COUNT OF ROTATIONS IN NORMAL SWEEP
630 C9=N*(N-1)/2
635 C8=C8+1
640 FOR J=1 TO N-1
650 FOR K=J+1 TO N
660 P=0\R=0\Q=0
700 FOR I=1 TO M
710 REM TRAP UNDERFLOW IF NOT SET TO ZERO
720 P=P+A(I,J)*A(I,K)
730 Q=Q+A(I,J)*A(I,J)
740 R=R+A(I,K)*A(I,K)
750 NEXT I
760 IF Q>=R THEN 800
770 REM EXCHANGE ROTATION
780 C=0\S=1
795 GOTO 860
800 IF Q*R<=T9 THEN 980 \ REM CHECK FOR SMALL COLUMNS
805 IF (P/Q)*(P/R)<=T9 THEN 980 \REM MAIN TEST OF UNNECESSARY ROTN
810 REM COMPUTE ROTN
820 Q=Q-R\V1=SQRT(4*P*P+Q*Q)
840 C=SQRT((V1+Q)/(2*V1))\ S=P/(V1*C)
860 REM ROTN - ON A THEN V
870 FOR I=1 TO M
880 R=A(I,J)\ A(I,J)=C*R+S*A(I,K)\ A(I,K)=-R*S+C*A(I,K)
910 NEXT I
920 FOR I=1 TO N
930 R=V(I,J)\ V(I,J)=C*R+S*V(I,K)\ V(I,K)=-S*R+C*V(I,K)
960 NEXT I
970 GOTO 1000 \ REM ROTN PERFORMED
980 REM COUNT SKIPPED ROTN BY DECREMENTING C9
990 C9=C9-1
1000 NEXT K
1010 NEXT J
1015 PRINT "END SWEEP ",C8,",",C9," ROTNS PERFORMED"
1020 IF C9>0 THEN 630 \ REM GO AGAIN IF NOT ALL ROTNS SKIPPED
READY
```

LISTING 1B

```
1030 C9=0 \ REM COUNT ZERO SINGULAR VALUES
1040 FOR J=1 TO N
1050 Q=0
1060 FOR I=1 TO M\ Q=Q+A(I,J)*A(I,J)\ NEXT I
1070 Z(J)=SQRT(Q) \ REM SAVE SINGULAR VALUE
1080 REM NOTE THAT SAVING S V IS NOT NECESSARY TO PROGRAM FUNCTION
1090 PRINT "SING. VAL.(",J,")=",SQRT(Q)
1100 PRINT "VECTOR (COL. OF MATRIX V)"
1110 FOR I=1 TO N\ PRINT V(I,J),\NEXT I
1120 PRINT
1130 REM MULTIPLY U BY S+ IMPLICITLY
1140 IF Q<=T9 THEN 1190
1150 REM SKIP IF SMALL SING VALUE
1160 FOR I=1 TO M\ A(I,J)=A(I,J)/Q \ NEXT I
1170 GOTO 1190
1180 C9=C9+1
1190 NEXT J
1200 IF C9>0 THEN PRINT C9," SINGULAR VALUES TAKEN AS ZERO"
1210 REM COMPLETE A+
1220 FOR I=1 TO N
1230 FOR J=1 TO M
1240 P=0
1250 FOR K=1 TO N \ P=P+V(I,K)*A(J,K)\ NEXT K
1260 B(I,J)=P \ REM GENERALIZED INVERSE
1270 NEXT J
1280 NEXT I
1290 RETURN \ REM COMPLETED TASK
1500 REM COPY A TO C
1510 FOR I=1 TO M
1520 FOR J=1 TO N \ C(I,J)=A(I,J) \ NEXT J
1530 NEXT I
1540 RETURN
2000 PRINT "RESIDUALS R =  Y - A * X"
2005 S = 0
2010 FOR I= 1 TO M
2020 R = Y(I)
2030 FOR J=1 TO N \ R=R-C(I,J)*X(J) \ NEXT J
2040 PRINT R, \ S=S+R*R
2060 IF 5*INT(I/5)=I THEN PRINT
2070 NEXT I
2080 PRINT \ PRINT "SUM OF SQUARES=",S
2090 RETURN
READY
```

Listing 2 presents the program Least Squares Driver (LSD). This, with 'geninv' appended, computes the generalized inverse of a matrix A, solves the least squares problem for the data Y fitted to A, and computes the residuals of this problem. The program uses the generalized inverse explicitly to compute the solution X. This is for clarity. Interested readers should find it straightforward to compute the solution from the decomposition as described above, thereby saving time and space.

LISTING 2

```
10 REM LEAST SQUARES DRIVER - LSD - J C NASH 1979
20 PRINT " LEAST SQUARES BY GENERALIZED INVERSE"
30 PRINT " TOLERANCE FOR ZERO",
40 INPUT T9
50 PRINT "M - NUMBER OF DATA POINTS",
60 INPUT M
70 PRINT "N - NUMBER OF EXPLANATORY VARIABLES",
80 INPUT N
90 DIM A(M,N),B(N,M),V(N,N),C(M,N),Z(N),Y(M)
110 PRINT "Y - VECTOR TO BE EXPLAINED"
120 FOR I=1 TO M\ INPUT Y(I) \ IF 5*INT(I/5)=I THEN PRINT\ NEXT I
130 REM
140 PRINT
```

```
160 PRINT "A - MATRIX OF VARIABLES"
170 FOR I=1 TO M
180 PRINT "ROW(",I,"):"
190 FOR J=1 TO N\INPUT1 A(I,J)\IF 5*INT(J/5)=J THEN PRINT\NEXT J
200 PRINT
210 NEXT I
250 GOSUB 1500 \ REM COPY A TO C
260 REM COMPUTE A+
270 GOSUB 500
275 GOSUB 400
280 REM FORM X = A+ * Y = B * Y
290 FOR I=1 TO N
300 P=0
310 FOR J=1 TO M \ P=P+B(I,J)*Y(J) \ NEXT J
320 X(I)=P
325 PRINT "X(",I,")=",P
330 NEXT I
370 REM RESIDUALS
380 GOSUB 2000
390 STOP
400 PRINT "GENERALIZED INVERSE B = A+"
410 FOR I=1 TO N
420 PRINT "ROW(",I,"):"
430 FOR J=1 TO M \ PRINT B(I,J),\IF 5*INT(J/5)=J THEN PRINT\NEXT J
440 PRINT \ NEXT I
450 RETURN
READY
```

Listing 3 gives the driver 'svdeig' which allows the singular value decomposition to be used to compute eigensolutions of a real, symmetric matrix. In my book I give a method for calculating the shift needed to make a matrix positive definite. Here, however, I rely on the calculation of residuals to verify that the eigenvalues have the correct sign.

LISTING 3

```
10 REM EIGENSOLUTION DRIVER SVDEIG - J C NASH 1979
20 PRINT "EIGENSOLUTIONS OF A SYMMETRIC MATRIX VIA "
25 PRINT "A SINGULAR VALUE DECOMPOSITION"
30 INPUT "ORDER=",N \ M=N
40 DIM A(N,N),V(N,N),B(N,N),C(N,N),Y(N),Z(N),X(N)
45 REM NOTE B NOT NEEDED EXCEPT FOR SUBROUTINE
60 PRINT "MATRIX A - ONLY LOWER TRIANGLE IS ENTERED"
70 FOR I=1 TO N
80 FOR J=1 TO I
90 PRINT "   A(",I,",",J,")=",
100 INPUT1 A(I,J)
110 IF 3*INT(J/3)=J  AND J<I THEN PRINT
115 A(J,I)=A(I,J)
120 NEXT J \ PRINT
130 NEXT I \ REM 70 TO 130 COULD SIMPLY CALCULATE MATRIX A
150 GOSUB 1500 \ REM COPY
170 PRINT "INPUT SHIFT (>1E35 STOPS PROGRAM)",
180 INPUT K9
190 IF K9>1E35 THEN STOP
195 INPUT "TOLERANCE FOR ZERO=",T9
200 FOR I=1 TO N
210 FOR J=1 TO N \A(I,J)=C(I,J) \ NEXT J
220 A(I,I)=C(I,I)+K9
230 NEXT I
270 REM SVD
280 GOSUB 500
290 REM EIGENVALUES BY SUBTRACTING SHIFT IF A IS POSITIVE DEFINITE
300 FOR I=1 TO N
310 Z(I)=Z(I)-K9 \ PRINT "EIGENVALUE(",I,")=",Z(I)
320 NEXT I
330 REM TEST
340 FOR K=1 TO N
350 PRINT "TEST ON EIGENVALUE ",K
360 FOR I=1 TO N
370 X(I)=V(I,K) \ Y(I)=Z(K)*X(I)
380 NEXT I
390 GOSUB 2000 \ REM RESIDUALS
400 NEXT K
410 GOTO 170
READY
```

Example solutions

Listing 4 presents a least squares fitting problem solved using LSD. This particular case is a polynomial least squares fit and the top of the listing shows how easily the programs may be altered to accommodate such tasks. It is also possible to make simple modifications to compute standard errors for the solution elements as well as to find other statistics.

LISTING 4

```
25 PRINT "MODIFIED TO PERFORM POLYNOMIAL LEAST SQUARES"
70 PRINT "DEGREE OF POLYNOMIAL TO BE FITTED (N-1)",
85 N=N+1
190 FOR J=1 TO N\A(I,J)=I^(J-1)\PRINT A(I,J),\NEXT J
RUN

 LEAST SQUARES BY GENERALIZED INVERSE
MODIFIED TO PERFORM POLYNOMIAL LEAST SQUARES
 TOLERANCE FOR ZERO?1E-8
M - NUMBER OF DATA POINTS?5
DEGREE OF POLYNOMIAL TO BE FITTED (N-1)?2
Y - VECTOR TO BE EXPLAINED
?4?9?16.1?24.9?37

A - MATRIX OF VARIABLES
ROW( 1):
 1 1 1
ROW( 2):
 1 2 4
```

```
ROW( 3):
 1 3 9
ROW( 4):
 1 4 16
ROW( 5):
 1 5 25
END SWEEP  1, 3 ROTNS PERFORMED
END SWEEP  2, 3 ROTNS PERFORMED
END SWEEP  3, 2 ROTNS PERFORMED
END SWEEP  4, 0 ROTNS PERFORMED
SING. VAL.( 1)= 32.156336
VECTOR (COL. OF MATRIX V)
 5.5272605E-02 .22444236 .97291859
SING. VAL.( 2)= 2.1977333
VECTOR (COL. OF MATRIX V)
 -.60228523 -.76967819 .21177339
SING. VAL.( 3)= .3743756
VECTOR (COL. OF MATRIX V)
 .79636513 -.59767975 9.2636208E-02
GENERALIZED INVERSE B = A+
ROW( 1):
 1.8000117 .00001102 -.79999261 -.59999803 .59999346

ROW( 2):
 -1.0571478 .32856275 .85713517 .52856846 -.65713631

ROW( 3):
 .14285762 -7.1427196E-02 -.1428558 -7.1428003E-02 .14285601

X( 1)= 1.580072
X( 2)= 1.375662
X( 3)= 1.1357224
RESIDUALS R =  Y - A * X
 -.0914564 .1257144 .17144 -.354278 .148558

SUM OF SQUARES= .20114244
STOP IN LINE 400
READY
```

Listing 5 shows the calculation of the generalized inverse of a singular matrix. Note the "zero" (very small) third singular value, and how the sum of squares of the residuals for the accompanying linear equation problem is non-zero.

LISTING 5

```
RUN

 LEAST SQUARES BY GENERALIZED INVERSE
 TOLERANCE FOR ZERO?1E-8
M - NUMBER OF DATA POINTS?3
N - NUMBER OF EXPLANATORY VARIABLES?3
Y - VECTOR TO BE EXPLAINED
?1?1?1
A - MATRIX OF VARIABLES
ROW( 1):
?1?1?1
ROW( 2):
?2?3?4
ROW( 3):
?5?6?7
END SWEEP  1, 3 ROTNS PERFORMED
END SWEEP  2, 3 ROTNS PERFORMED
END SWEEP  3, 2 ROTNS PERFORMED
END SWEEP  4, 0 ROTNS PERFORMED
SING. VAL.( 1)= 11.896793
VECTOR (COL. OF MATRIX V)
 .45815815 .57007348 .68198777
SING. VAL.( 2)= .68287635
VECTOR (COL. OF MATRIX V)
 .78957222 .09137804 -.60681608
SING. VAL.( 3)= 5.1090704E-06
VECTOR (COL. OF MATRIX V)
 -.40824846 .81649634 -.40824859
GENERALIZED INVERSE B = A+
ROW( 1):
 .46969984 -.9545375 .45456224
ROW( 2):
 6.0606012E-02 -9.0910628E-02 .09090738
ROW( 3):
 -.34848603 .77272172 -.27273648
X( 1)= -.03027542
X( 2)= 6.0602764E-02
X( 3)= .15149921
RESIDUALS R =  Y - A * X
 .81817343 .27274567 -.27273398
SUM OF SQUARES= .81818178
STOP IN LINE 400
READY
```

Listing 6 solves the linear equation problem and performs a matrix inversion for a well-behaved matrix.

Listing 7 uses the driver 'svdeig' to compute the eigenvalues of a simple matrix. Note that the first try fails because the shift (0.0) leaves a matrix which is not positive definite.

These programs and methods are useful in many areas of practical computation. They are reliable and numerically sound when used properly. Because of the generality of the generalized inverse routine 'geninv', it is not necessarily the most efficient technique for all the tasks it can perform, either in memory space or in computer time. Nevertheless, this one tool can provide microcomputer users with the power to solve a large number of real-world problems with little fuss and bother. □

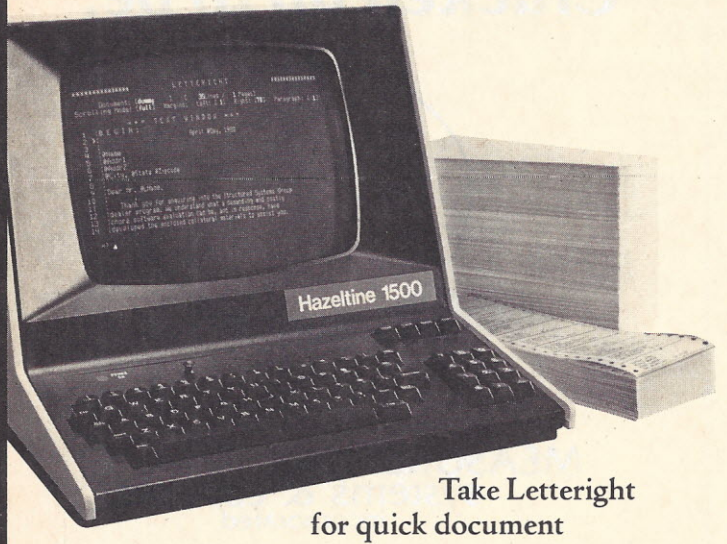
LISTING 6

```
RUN

 LEAST SQUARES BY GENERALIZED INVERSE
 TOLERANCE FOR ZERO?1E-8
M - NUMBER OF DATA POINTS?1_3
N - NUMBER OF EXPLANATORY VARIABLES?3
Y - VECTOR TO BE EXPLAINED
?1?1?1
A - MATRIX OF VARIABLES
ROW( 1):
?1?1?1
ROW( 2):
?1?2?2
ROW( 3):
?1?2?3
END SWEEP  1, 3 ROTNS PERFORMED
END SWEEP  2, 3 ROTNS PERFORMED
END SWEEP  3, 2 ROTNS PERFORMED
END SWEEP  4, 1 ROTNS PERFORMED
END SWEEP  5, 0 ROTNS PERFORMED
SING. VAL.( 1)= 5.0489175
VECTOR (COL. OF MATRIX V)
 .32797765 .59100567 .73698234
SING. VAL.( 2)= .64310412
VECTOR (COL. OF MATRIX V)
 .73697969 .32799131 -.59100138
SING. VAL.( 3)= .30797851
VECTOR (COL. OF MATRIX V)
 -.59100897 .73697625 -.32798535
GENERALIZED INVERSE B = A+
ROW( 1):
 2.0000338 -.99994414 .00006443
ROW( 2):
 -.99997971 2.0000273 -.99997566
ROW( 3):
 -.0000154 -1.0000397 .99993913
X( 1)= 1.0001541
X( 2)= .00007194
X( 3)= -.00011597
RESIDUALS R =  Y - A * X
 -.00011007 -.00006604 .00004993
SUM OF SQUARES= 1.8969692E-08
STOP IN LINE 400
READY
```

LISTING 7

```
EIGENSOLUTIONS OF A SYMMETRIC MATRIX VIA
A SINGULAR VALUE DECOMPOSITION
ORDER=3
MATRIX A - ONLY LOWER TRIANGLE IS ENTERED
   A( 1, 1)=?1
   A( 2, 1)=?1   A( 2, 2)=?0
   A( 3, 1)=?0   A( 3, 2)=?1   A( 3, 3)=?-1
INPUT SHIFT (>1E35 STOPS PROGRAM)?0.0
TOLERANCE FOR ZERO=1E-8
END SWEEP  1, 2 ROTNS PERFORMED
END SWEEP  2, 1 ROTNS PERFORMED
END SWEEP  3, 1 ROTNS PERFORMED
END SWEEP  4, 0 ROTNS PERFORMED
SING. VAL.( 1)= 1.7320509
VECTOR (COL. OF MATRIX V)
 .40824826 -.40824828 .81649659
SING. VAL.( 2)= 1.7320508
VECTOR (COL. OF MATRIX V)
 -.70710678 -.70710676 0
SING. VAL.( 3)= 2.7688747E-08
VECTOR (COL. OF MATRIX V)
 .57735026 -.57735027 -.57735025
EIGENVALUE( 1)= 1.7320509
EIGENVALUE( 2)= 1.7320508
EIGENVALUE( 3)= 2.7688747E-08
TEST ON EIGENVALUE  1
RESIDUALS R =  Y - A * X
 .70710679 -1.9318517 2.6389586
SUM OF SQUARES= 11.196154
TEST ON EIGENVALUE  2
RESIDUALS R =  Y - A * X
 .18946864 -.51763802 .70710676
SUM OF SQUARES= .80384746
TEST ON EIGENVALUE  3
RESIDUALS R =  Y - A * X
 .00000002 -.00000003 .00000001
SUM OF SQUARES= 1.4E-15
INPUT SHIFT (>1E35 STOPS PROGRAM)?3
TOLERANCE FOR ZERO=1E-8
END SWEEP  1, 3 ROTNS PERFORMED
END SWEEP  2, 3 ROTNS PERFORMED
END SWEEP  3, 0 ROTNS PERFORMED
SING. VAL.( 1)= 4.7320507
VECTOR (COL. OF MATRIX V)
 .78865764 .57736778 .21134229
SING. VAL.( 2)= 2.9999999
VECTOR (COL. OF MATRIX V)
 -.57737414 .57733281 .57734383
SING. VAL.( 3)= 1.2679492
VECTOR (COL. OF MATRIX V)
 .21132487 -.5773502 .78867515
EIGENVALUE( 1)= 1.7320507
EIGENVALUE( 2)= -.0000001
EIGENVALUE( 3)= -1.7320508
TEST ON EIGENVALUE  1
RESIDUALS R =  Y - A * X
 -.00003042 .00003037 .00003007
SUM OF SQUARES= 2.7519182E-09
TEST ON EIGENVALUE  2
RESIDUALS R =  Y - A * X
 .00004139 .00003025 .00001096
SUM OF SQUARES= 2.7483162E-09
TEST ON EIGENVALUE  3
RESIDUALS R =  Y - A * X
 -.00000008 -.00000014 -.00000005
SUM OF SQUARES= 2.85E-14
INPUT SHIFT (>1E35 STOPS PROGRAM)?1E36
STOP IN LINE 195
READY
```

SSG Writing and Mailing Systems.



Take Letteright for quick document preparation and edit plus NAD Name And Address for extensive mailing list capabilities.

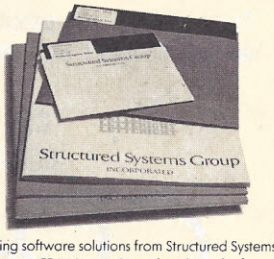
Put them together and you've got a flexible, powerful solution to big and small correspondence problems.

With Letteright you create and edit your document right on the screen. It's much easier to use than a typewriter. The letters are always perfect, and revisions are a snap.

Letteright's "wild card" slots let you create standard letters and forms, then insert information selected from your mailing list to address and "personalize" the letter.

The NAD system will store lots of names and addresses, with identifying information you create. You then print lists, labels, or envelopes of virtually any group you want from the list, or the whole list.

This pair should be working for every microcomputer owner.



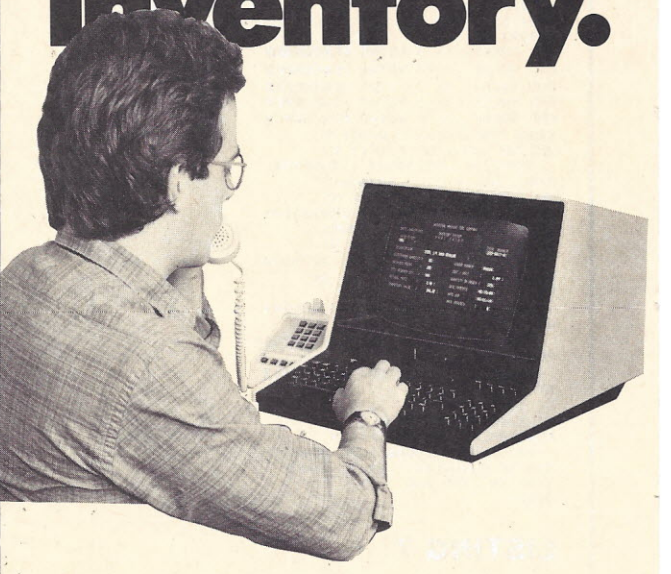
Letteright and NAD are part of a full line of working software solutions from Structured Systems Group, all ready to run on any CP/M® microcomputer system. CP/M is a registered trademark of Digital Research.

CIRCLE INQUIRY NO. 60

Structured Systems

5204 Claremont Oakland, Ca 94618 (415) 547-1567

The Working Inventory.



Here's a straightforward stock control solution that will replace stacks of paperwork while helping you control your expensive inventory.

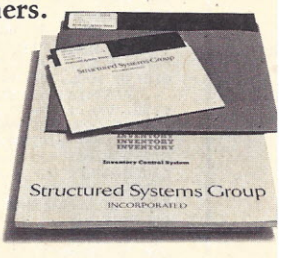
Today's money costs make inventory carrying costs more burdensome than ever. And the cost of inventory control personnel isn't exactly going down, either.

Now you can enter your stock additions and depletions once, at the keyboard of your microcomputer.

You'll get automatic re-order reports (to show you what you need), complete inventory parts listings and valuations (to show you what you have), and activity reports (to show what's moving).

You'll worry less, because you'll know more. The Inventory Control System includes an auditability option, extensive error screening, and reliable documentation. It was designed for people, not for programmers.

Call or write SSG for a complete System Summary and the names of your closest dealers. There are over 150 nationwide.

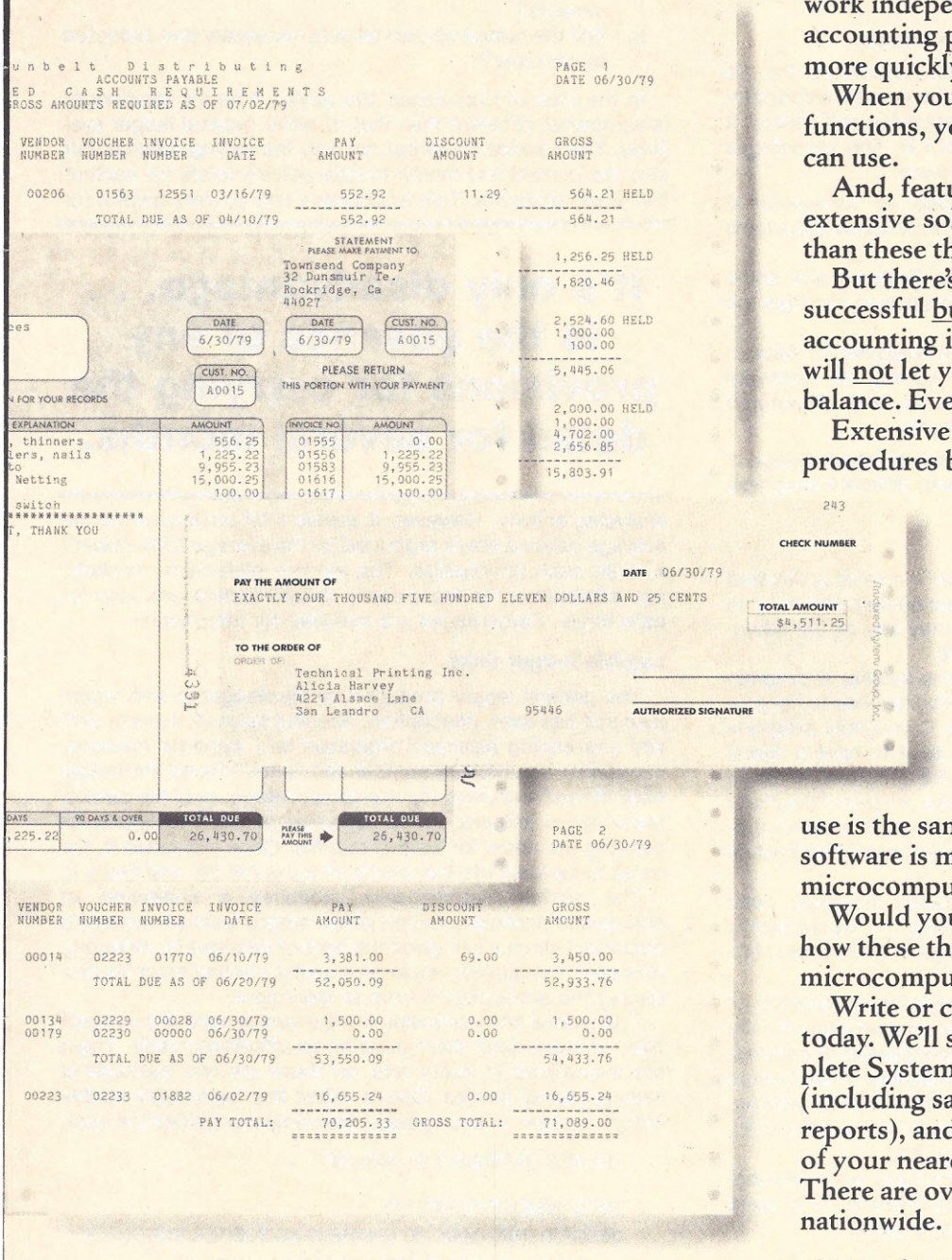


The Inventory Control System is part of a full line of working software solutions from Structured Systems Group, all ready to run on any CP/M® microcomputer system. CP/M is a registered trademark of Digital Research.

Structured Systems

5204 Claremont Oakland, Ca 94618 (415) 547-15

The Working General Ledger, Accounts Receivable, Accounts Payable.



unbelt Distributing — PAGE 1
ACCOUNTS PAYABLE — DATE 06/30/79
E D C A S H R E Q U I R E M E N T S
ROSS AMOUNTS REQUIRED AS OF 07/02/79

VENDOR NUMBER	VOUCHER NUMBER	INVOICE NUMBER	INVOICE DATE	PAY AMOUNT	DISCOUNT AMOUNT	GROSS AMOUNT
00206	01563	12551	03/16/79	552.92	11.29	564.21 HELD
	TOTAL DUE AS OF 04/10/79			552.92		564.21

STATEMENT
PLEASE MAKE PAYMENT TO:
Townsend Company
32 Dunsmuir Te.
Rockridge, Ca
44027

DATE 6/30/79 | DATE 6/30/79 | CUST. NO. A0015
CUST. NO. A0015
PLEASE RETURN THIS PORTION WITH YOUR PAYMENT

EXPLANATION	AMOUNT	INVOICE NO.	AMOUNT
, thinners	556.25	01555	0.00
lers, nails	1,225.22	01556	1,225.22
to	9,955.23	01583	9,955.23
Netting	15,000.25	01616	15,000.25
	100.00	01617	100.00

switch
T, THANK YOU

1,256.25 HELD
1,820.46
2,524.60 HELD
1,000.00
100.00
5,445.06
2,000.00 HELD
1,000.00
4,702.00
2,656.85
15,803.91

243
CHECK NUMBER
DATE 06/30/79
PAY THE AMOUNT OF
EXACTLY FOUR THOUSAND FIVE HUNDRED ELEVEN DOLLARS AND 25 CENTS
TOTAL AMOUNT $4,511.25
TO THE ORDER OF
Technical Printing Inc.
Alicia Harvey
4221 Alsace Lane
San Leandro CA 95446
AUTHORIZED SIGNATURE

DAYS	90 DAYS & OVER	TOTAL DUE	PLEASE PAY THIS AMOUNT	TOTAL DUE
,225.22	0.00	26,430.70		26,430.70

PAGE 2
DATE 06/30/79

VENDOR NUMBER	VOUCHER NUMBER	INVOICE NUMBER	INVOICE DATE	PAY AMOUNT	DISCOUNT AMOUNT	GROSS AMOUNT
00014	02223	01770	06/16/79	3,381.00	69.00	3,450.00
	TOTAL DUE AS OF 06/20/79			52,050.09		52,933.76
00134	02229	00028	06/30/79	1,500.00	0.00	1,500.00
00179	02230	00000	06/30/79	0.00	0.00	0.00
	TOTAL DUE AS OF 06/30/79			53,550.09		54,433.76
00223	02233	01882	06/02/79	16,655.24	0.00	16,655.24
			PAY TOTAL:	70,205.33	GROSS TOTAL:	71,089.00

Three proven, refined software packages that work independently or together to simplify your accounting procedures, produce statements more quickly, and track information better.

When you computerize these important functions, you naturally look for features you can use.

And, feature for feature, you won't find more extensive solutions running on microcomputers than these three systems.

But there's a single feature essential to the successful business system. Reliability. SSG accounting is reliable. For example, the systems will not let your operator or files get out of balance. Ever.

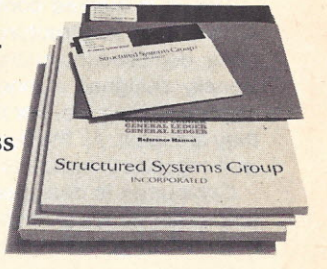
Extensive error-checking and file back-up procedures built into the software reduce errors, and protect your valuable time and data.

Enter your transaction once, and file it. Thorough audit trails ensure data integrity from source document to financial statement, as the computer quickly and accurately finishes your bookkeeping job.

This reliability is guaranteed, because the software system you use is the same as the master copy at SSG. Your software is maintained and updated by a leading microcomputer software vendor.

Would you like more details and specifics on how these three software systems can put a microcomputer to work for your business?

Write or call SSG today. We'll send a complete System Summary (including sample reports), and the address of your nearest dealers. There are over 150 nationwide.

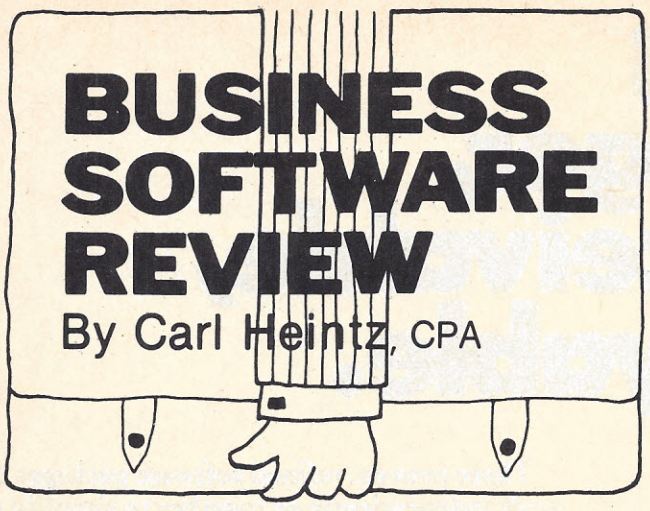


G.L., A/R, and A/P are all part of a full line of working software solutions from Structured Systems Group, all ready to run on any CP/M® microcomputer system. CP/M is a registered trademark of Digital Research.

CIRCLE INQUIRY NO. 61

Structured Systems Group INCORPORATED

5204 Claremont Oakland, California 94618 (415) 547-1567



Maxiledger: Tailored to Tastes

Maxiledger, produced by Compumax Associates, Inc., is advertised as a low cost competitor to the general ledger programs sold by Structured Systems, Peachtree and others. It is written in Micropolis Basic, release 1.0. The advantages claimed for this series of programs include:

1. Price—Maxiledger sells for $350 as compared to $1000 for Peachtree and $995 for Structured Systems software.
2. Source code—unlike Structured Systems and some other general ledger systems, Compumax provides the user with the source code.
3. Hardware options—Compumax is designed for Apples, TRS-80, CBasic/CP/M and a variety of hardware configurations, whereas Peachtree and Structured Systems are designed for CP/M only.

With all of those advantages, what are its disadvantages? As in the case of any computer software, different programs are best suited for different users.

Customize your programs

Approximately one-third of Maxiledger's manual is devoted to the program listings, which are documented and easy to read. As an example of programming, they are, as the advertisements state, "eloquent" yet simple.

It is obvious that the user interested in looking at program listings will be interested in making customizations to the programs. As an aid to this process, the Compumax programmers have taken the attitude that it's best to have a liberal use of remark statements throughout the programs so as to simplify the job. Add to that the extensive program flowcharts (some of the best I have seen) and you've got a set of programs that provide the basis to most any kind of modification a user may wish to attempt.

The marketing philosophy of Compumax stresses: "with Compumax software, you have a beginning. . .it's your turn to tell the computer how to run the business. . .Compumax software is designed with change in mind."

This is in marked contrast to the marketing philosophy of others, such as Structured Systems. Its programs come complete, compiled (no changes possible) and packed in a professionally designed manner. They are designed to be booted and run as they are. For the user unfamiliar with programming, they are ideal. All the bugs have been taken out and the slick, well-organized manual is an excellent guide.

Compumax is designed for those who have some elementary understanding of the programming process, or at least will want to get into that process at some point.

Variations in system design

The Maxiledger system uses some of the same principles of program architecture as does Structured Systems—the account number defines where the amounts will appear on the financial statements. Unfortunately, the system is not as flexible as Structured Systems'. For example, in the Structured Systems GL, one may have up to ten total categories of assets; under the Maxiledger scheme, there are only two: current and noncurrent. Of course, with a little bit of programming, this would be expanded. The same rigidity applies to the liabilities—again with only two main divisions existing.

On a financial statement, Structured Systems gives the user the option of summarizing several detail accounts into one total which appears on the financials as a single figure. In the Compumax scheme, the detail figures will appear subtotaled with a subtotal account title. So much for charts of accounts.

The user who wants to "plug and run" with a general ledger program would do well to look at the output of Compumax and other general ledger systems before purchase. In the end, there are only a few factors that determine whether a system is satisfactory:

1. Will the output be readable and in the form the user expects?
2. Will the output contain all data necessary and expected by the user?

In the case of Compumax, the output from the Maxiledger is somewhat different than that of other general ledger systems. The balance sheet has not only the ending balance, but also the current and month-to-date activity totals for each of the asset accounts. This information can be very helpful for analyzing activity. However, it seems a bit unusual since the average balance sheet produced by the average CPA doesn't contain such information. The income statement, similarly, has information for the current, month-to-date and year-to-date totals. Percentages are included for information.

It's only disadvantage. . . is the absence of any provisions for carrying the detail for several months.

Legible ledger links

The general ledger produced by Maxiledger is well organized and has date, description, opening balance, current activity and ending balances organized into separate columns, capped by the account number and name. I found the ledger easy to read and clear. Its only disadvantage, as in the case of many micro ledgers, is the absence of any provisions for carrying the detail for several months. Once you close it, the detail is merged into the charts of accounts file and that's it.

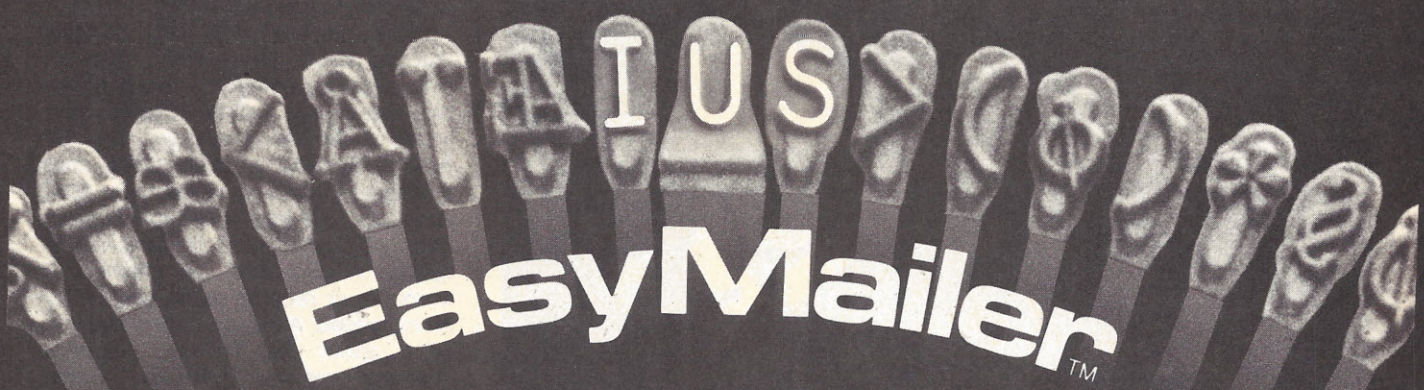
The system automatically produces a statement of changes in financial position which differs from an account's statement since it just gives the total of changes by category. However, the system would be a useful starting point for producing the accountant's form of statement.

The journal entry process is organized around the concept that for every debit there must be an offsetting credit. There are exceptions to every rule, so there are two methods of making journal entries. One is called the "automatic double entry" system. In this system, the computer guides the user:

```
DOUBLE ENTRIES # 31 AND 32

DATE (MMDDYY)? 052179
DESCRIPTION (MAX. 30 CHARS.)? SORCERER PURCHASE
AMOUNT (NO MINUS SIGN, PLEASE)?  1295.45
DR ACC'T #? 8315
CR ACC'T #? 1011
ANOTHER DOUBLE ENTRY   Y OR N? Y
```

EasyWriter™

The Professional Word Processing System for your Apple-II Personal Computer



EasyMailer™

A Continuous Letter Writer

EasyMover™

Personal Electronic Mail

IUS (Information Unlimited Software, Inc.), 281 Arlington Ave., Berkeley, CA 94707 415-525-4046 / 525-9452

e is a TM of Apple Computers, Inc.

CIRCLE INQUIRY NO. 29

EasyWriter is a TM of Cap'n Software.

In most cases, this entry sequence will work satisfactorily. Unfortunately, however, there are times when an entry will affect more than one account. The "automatic double entry" system does not work in this case, and the user will have to choose from the other option—"single journal entries." In this method, the user does not concern himself with D_r or C_r, only whether an account has been increased or decreased. This can be somewhat confusing for an accountant who is used to thinking in terms of D_r and C_r, but for a user who is unfamiliar with accounting for the debits and credits, it might provide less confusion.

Maxiledger also includes a short routine called "updating existing journal records" which functions as an editor to a file of journal entries. With this function a user may insert new journal records, change data or delete entire records. It is followed by a menu which allows all journal entries to be listed. The listing routines provide the user with proof that the debits equal the credits and that the journal file is complete before posting. Only one journal file is prepared—in other words, in contrast to many manual systems in which there are several types of journals, such as the cash journal, the receipts journal and the sales journal—in the Maxiledger system, there is only one journal. (This is the same for other systems such as Structured Systems for example.)

Two entry speeds possible

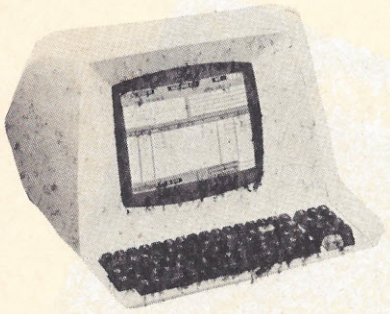
One of the most unique features which the Maxiledger system has is the ability to choose between two types of posting—slow posting or fast posting. At first glance, everyone would seem to want fast posting—after all, who wants to go slow? The differences in the two methods, however, can have profound impact upon whether the posting process can be accomplished.

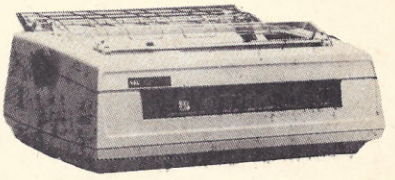
If the user has a 48K CPU, and let's assume 1000 transactions, then the slow posting method is the only way that the system will be able to cope. Most of the time, however, the fast posting system is used since small businesses usually have less than 1000 accounts and less than 250 transactions as a posting. If there is a 64K system in use, then the numbers of accounts or records to be posted can be increased. Although the manual cautions about the ominous message 'memory overflow,' there are no solutions offered as to how to avoid it or, more importantly, what to do when confronted with it. (Some micro users have nightmares about nasty things like memory overflows.)

So how does Maxiledger stack up? A CPA who must turn out financial statements for clients and needs rapid data entry probably would not find Maxiledger as suitable as another system. The small businessman with a need to handle the books with a minimum of hassle will find Maxiledger useful. To fully appreciate the system, however, Maxiledger should be in the hands of a user who has some familiarity with microcomputers, and is not intimidated by flowcharts and basic programs. A competent programmer can easily customize the system to meet his needs. □

AUTOMATED EQUIPMENT INCORPORATED

Toll-Free Ordering
1-800-854-6003 Outside Calif.
714-739-4701 Inside Calif. Outside Cont. USA





TERMINALS

Televideo 912-B	$750
Televideo 912-C	775
Televideo 920-B	810
Televideo 920-C	835
Soroc IQ-120	740
Zenith Z-19	850
Hazeltine 1410	750
Hazeltine 1420	830
Hazeltine 1500	950
Hazeltine 1510	1050

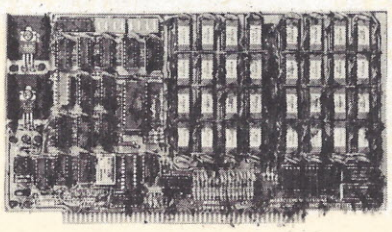
NORTHSTAR

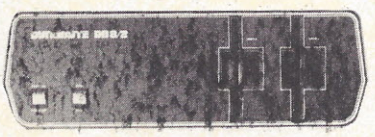
	KIT	ASM
HRZ-1-32K-D	$1575	$2100
HRZ-2-32K-D	1875	2400
HRZ-1-32K-Q	1800	2450
HRZ-2-32K-Q	2300	2800
Additional 16K RAM	325	365
Additional 32K RAM	515	565
HARD DISC SYSTEM		3950

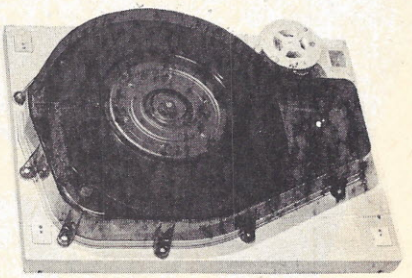
DB6400 64K RAM MAY BE SUBSTITUTED FOR ASM 32K NORTHSTAR FOR $150. NORTHSTARS WITHOUT MEMORY AVAILABLE.

PRINTERS

NEC 5510	$2700
NEC 5520	2975
MALIBU	2295
TI-810	1580
TI-820	1580
EPSON	CALL
OKIDATA	CALL
ANADEX DP-8000	795
ANADEX DP-9500	1350
BASE 2 800	600







BOARDS

	KIT	ASM
CB2 Z80CPU	$168	$220
VB1B VIDEO	125	170
VB2 VIDEO	135	188
VB3 VIDEO	300	347
IO4 INTERFACE	135	188
SBI SYNTHESIZER	161	227

MEASUREMENT SYSTEMS

DM3200	500
DM6400	640
DMB3200	650
DMB6400	790

DYNABYTE

DB 8/1 48K	$2395
DB 8/1 64K	2715
DB 8/2 48K	3900
DB 8/2 64K	4200
DB 8/4 SINGLE	3030
DB 8/4 DUAL	3830
10M WINCHESTER	4630
32M PHOENIX	11800
64M PHOENIX	15000
96M PHOENIX	18000

MORROW

DISCUS-1 1 DRIVE	$ 815
DISCUS-1 2 DRIVE	1465
DISCUS 2D 1 DRIVE	980
DISCUS 2D 2 DRIVE	1635
DISCUS 2 + 2 1 DRIVE	1265
DISCUS 2 + 2 2 DRIVE	2245
DISC JOCKEY 1	190
DISC JOCKEY 2D	350
DISCUS M26 HARD DISC	4095
ADDITIONAL HARD DISC	3685

SOFTWARE

CPM-2	$ 150
CBASIC-2	100
WORDSTAR	350
MPM	300
N.S. PASCAL	160
WHATSIT	170
SELECTOR C-III	280

We will deduct an additional 10% from our already discounted software prices when you purchase a system from us.

GRAHAM-DORIAN

ACCTS. REC	$950	INVEN	$550
ACCTS. PAY	950	CASH REG	550
GEN. LEDG	950	PAYROLL	550
JOB COST	950	APARTMENT	550

MISC

ATARI 400	$450
ATARI 800	840
TI 99-4	CALL
RS 232 CABLE 5′	20
RS 232 CABLE 10′	25
LEDEX MONITOR	125
NOVATION CAT	170

DISCS-BOX OF 10

VERBATUM 5¼	1 SIDE	$28
VERBATUM 5¼	2 SIDE	45
VERBATUM 8	1 SIDE	35
VERBATUM 8	2 SIDE	55
5″ PLASTIC BOX		$3.00
8″ PLASTIC BOX		3.50

We will try to beat any advertised price.

Automated Equipment Inc.
Suite #D
4341 W. Commonwealth Ave.
Fullerton, Calif. 92633

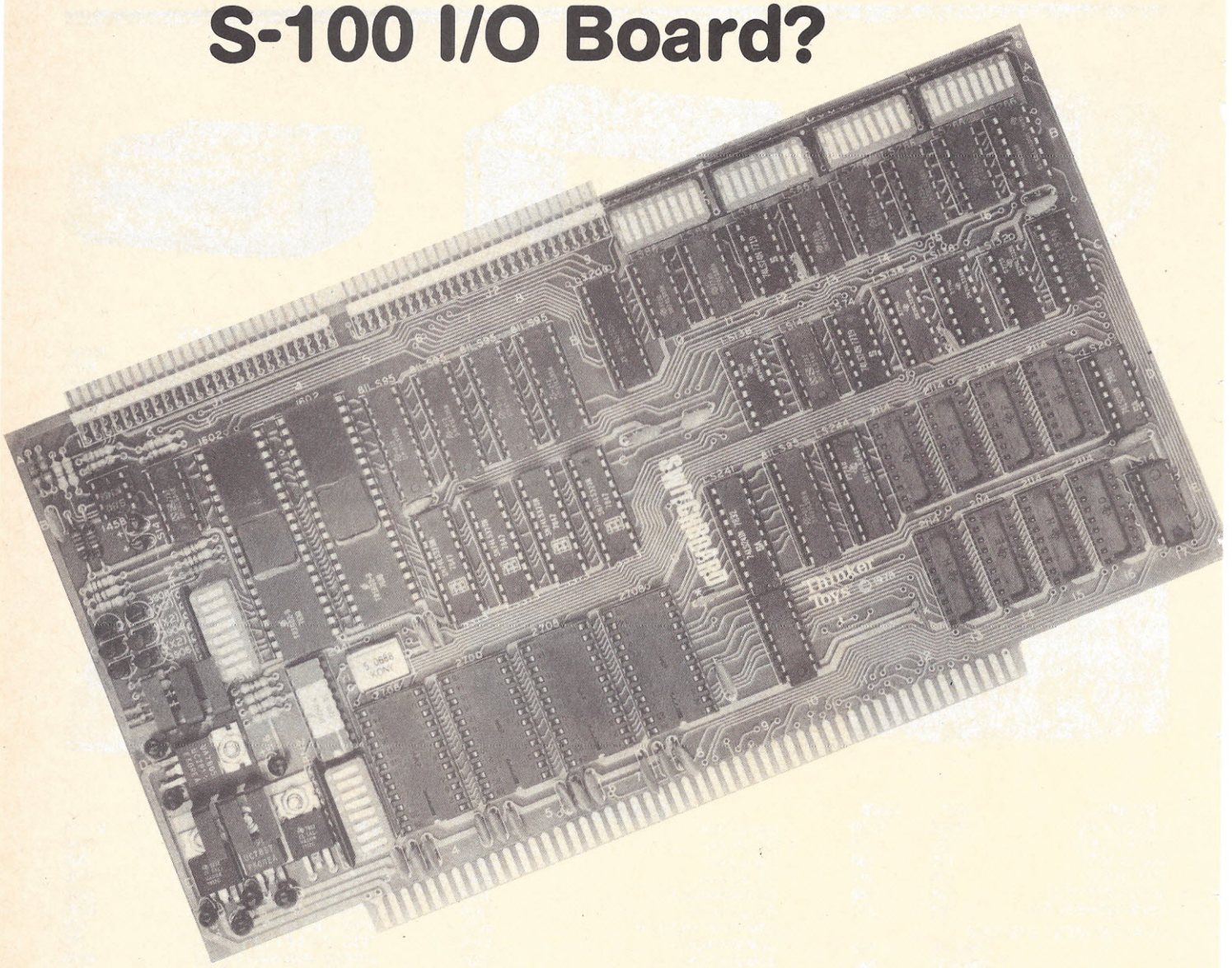
OPEN MON-FRI 8 AM TO 5 PM

TERMS: All prices listed are cash discounted and are subject to change or withdrawal. Credit cards and COD's are accepted at 2% additional handling charge. Universities and Well Rated Firms NET-10.

SHIPPING: ADDITIONAL IN ALL CASES.

TECH. ASSISTANCE: WE TRY TO HELP INTERFACE AND TROUBLESHOOT CALL 714-739-4703

The Ultimate S-100 I/O Board?



by Roger Edelson

With the maturing of the microcomputer industry, there has been a diminished effort in S-100 bus kits or systems built from a conglomerate of separate boards. Most new entrants into personal or small business computing are starting with complete systems, and those early pioneers and kit builders have gone on to applications.

It might seem somewhat inappropriate at this time to review a new I/O board, but many users of mature systems are finding they have a need for additional ports, or are seeking ways of freeing up some slots on their now cramped S-100 motherboard. It is these two problems that the Thinker Toys Switchboard addresses. . . and addresses well.

Switchboard combines eight I/O ports, 4K of Eprom space, and 4K of static RAM all on one standard S-100 board compatible with the proposed IEEE S-100 bus standard. The I/O portion of the board is essentially configured as a 4P + 2S + 2Sp (4-parallel, 2-serial, and a status and a strobe port—the partridge in a pear tree is an optional accessory). The inclusion of 4K of Eprom is particularly useful for storing small programs needed on power-up or disk bootstrap operations. The 4K of RAM nicely fills up the rest of the 8K address space making system partitioning easier. The Rev. 2 modification to the Switchboard adds the provision of insertable wait states—a useful advantage over the older model when operating in the newer, faster systems.

The serial ports are comprised of two 1602 UARTs (universal asynchronous receiver transmitters) with individually switch-selectable baud rates. There are 16 different selectable baud rates ranging from 50 to

Table 1. Designating Parallel Ports as Input or Output ("on = OUTPUT "off" = INPUT)

I/O PORT BASE + 4	I/O PORT BASE + 5	I/O PORT BASE + 6	I/O PORT BASE + 7
PARALLEL PORT#1	PARALLEL PORT#2	PARALLEL PORT#3	PARALLEL PORT#4 (pulled up)
SW4-3	SW4-2	SW4-1	SW4-4

19,200 and covering all the major I/O transmission rates in between. One 8-switch dip switch is used to select these rates. Additional dip switches are used to configure the serial data word length, stop bit count, and parity select. The word length cannot be switch programmed to 5 or 6 bits, although this capability is present in the Western Digital 1602. Thinker Toys indicates that trace modifications can be made to obtain the 5 data-bit and 1½ stop bit word necessary to interface with Baudot devices. They, however, do not recommend this modification.

Four selected status lines used

The serial status port provides outputs of four selected status lines from each of the 1602 UARTs. The low nibble is used to provide the status information for serial device 1, and the high nibble gives the same status for serial device 2. The status lines echoed by the status port are transmitter buffer empty, over-run error, parity error, and data ready. Data ready and transmitter buffer empty are two signals required for correct handshaking operation of the UART. The over-run error occurs if the DRR (data received reset) line has not been taken low before another serial word is received. This error occurs if the CPU was unable to accept a data word before a new one has arrived in the UART receiver register.

A parity error occurs if the received word does not match the selected parity of the UART. If parity has been disabled, no error will occur, no matter which parity (odd or even) is received. A status flag not provided in the status port is framing error. This error signal occurs if the received character has no valid stop-bit. This only occurs in the case of a malfunction of one

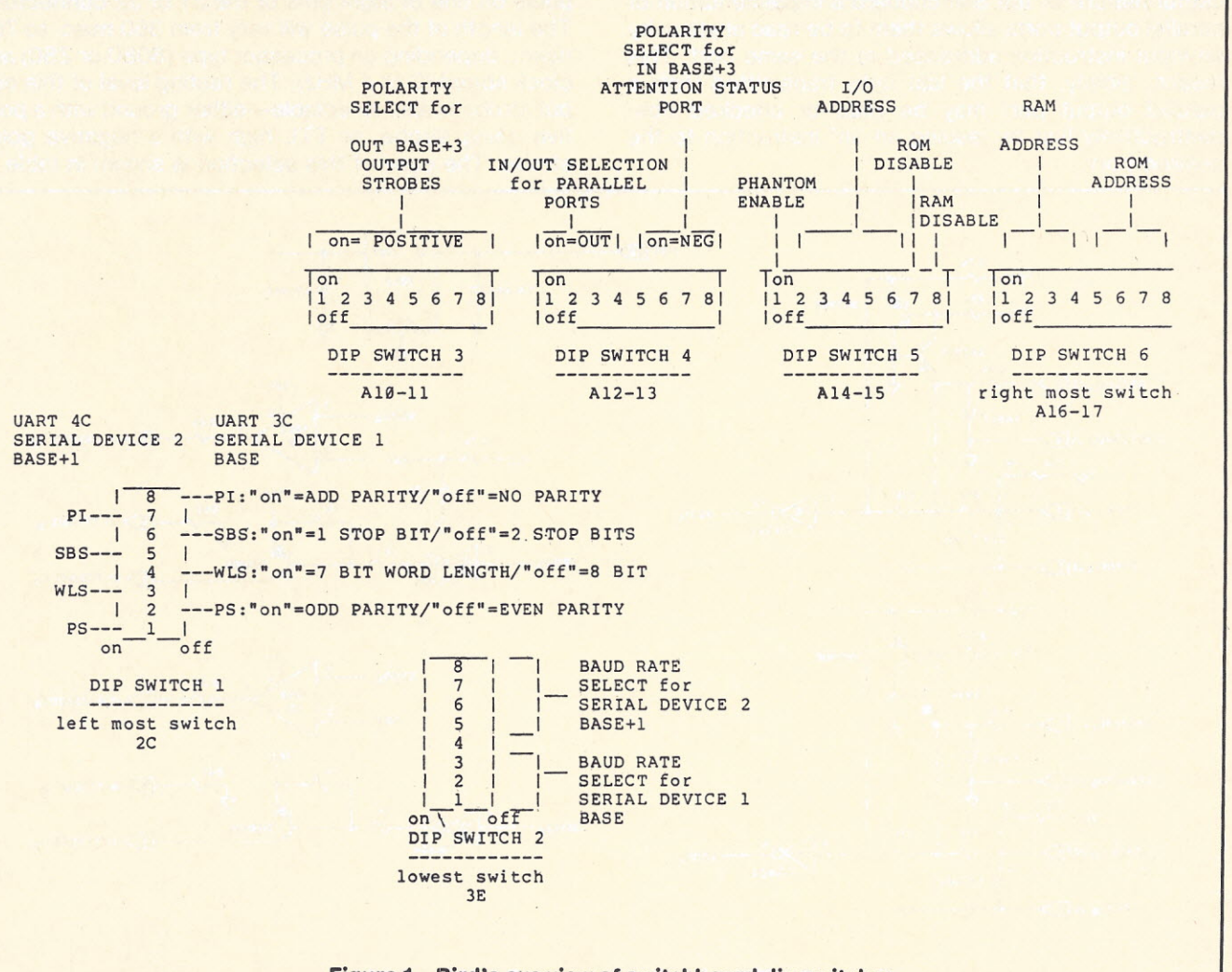


Figure 1. Bird's eye view of switchboard dip switches and their functions.

Table 2 Relation between the polarity switches on SW4 and their respective attention port bit.

ATTENTION PORT BASE + 3 AND POLARITY SWITCHES
("on" = Positive to Negative Transition)
("off" = Negative to Positive Transition)

BIT 3 . . . . . SW 4-6	BIT 2 . . . . . SW 4-7	BIT 1 . . . . . SW 4-5	BIT 0 . . . . . SW 4-8

of the pair of communicating UARTs, or a failure in the interconnecting cable.

As these types of failures are not CPU dependent, there is no real need to monitor this status flag during normal serial data transmission. The Switchboard is designed to provide a data received F/F reset signal every time the serial device is addressed so there is no reset overhead operation for the user to worry about.

As mentioned earlier, the Switchboard provides four 8-bit parallel ports that may be assigned either input or output status by setting the appropriate dip switch. This switch assignment is shown in table 1. When set for an output port configuration, the parallel output bits are strobed into an octal latch (LS373s). Because the data has been latched, it remains at that output port until changed by a subsequent 'out' instruction. This very useful feature of the Switchboard's implementation of parallel output ports allows them to be read anytime by an input instruction addressed to the same port. This means, simply, that the last byte transmitted from a parallel output port may be read or checked non-destructively just by issuing an 'in' instruction to the desired port.

Unification of peripherals

In addition to its status port, the Switchboard provides an input attention port that may be set by selected peripheral strobes. Thinker Toys has included this in its design switches to establish which transition polarity of the peripheral strobe the attention port will recognize. This feature is extremely useful because it is seldom possible to arrange all peripherals to have the same transition edge polarity.

Also built into the parallel port/attention port interface is the capability for the automatic reset of the attention/strobe bit when an 'in' instruction to the associated parallel port is executed.

Thinker Toys has included an output strobe port which, when properly addressed, will put out a single pulse on one of eight pins of the J1 or J2 connectors. The length of the pulse will vary from 350 nsec. to 700 nsec., depending on processor type (8080 or Z80) and clock speed (2 or 4 MHz). The resting level of this output strobe is also selectable—either ground with a positive going strobe, or TTL high with a negative going strobe. The ease of this selection is shown in table 2.

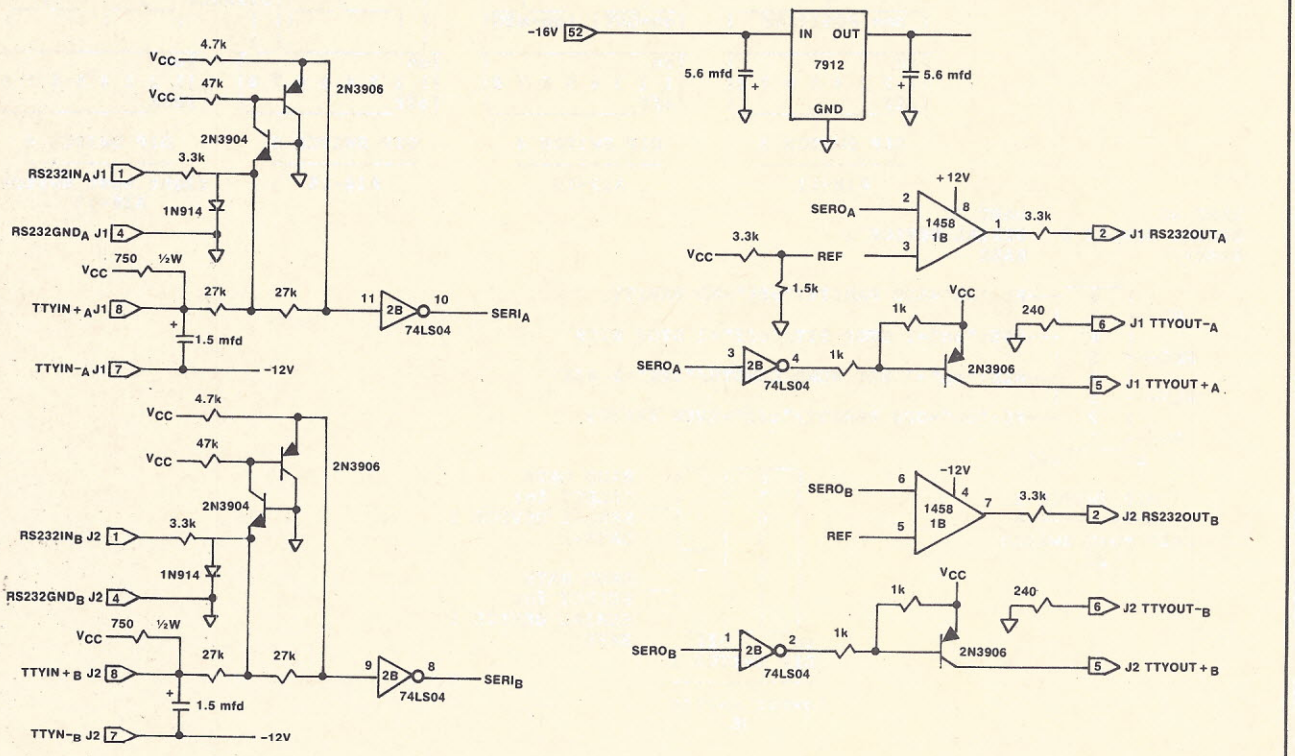
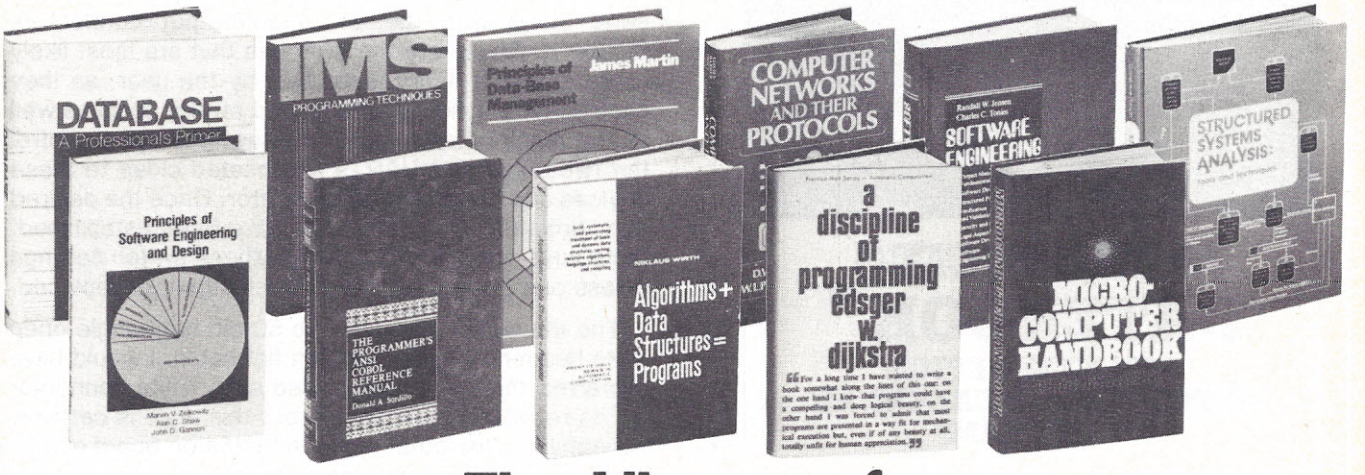


Figure 2. TTY circuit loop and RS-232 interface design.

Discover

the most organized (and inexpensive) way to keep up with what's new in computer techniques and management—



The Library of Computer and Information Sciences

(Publishers' Prices shown)

41625. DATABASE: A PROFESSIONAL'S PRIMER. *David Kroenke.* Covers every aspect of data base concepts, systems and applications. **$24.95**

41785. DEBUGGING SYSTEM 360/370 PROGRAMS USING OS AND VS STORAGE DUMPS. *D. H. Rindfleisch.* Guide to storage dump debugging. Illustrations, examples, sample dumps. **$19.95**

42303. A DISCIPLINE OF PROGRAMMING. *Edsger W. Dijkstra.* New programming tools to solve problems that range from the everyday to the complex. **$21.95**

80837. STRUCTURED SYSTEMS ANALYSIS: Tools and Techniques. *Gane and Sarson.* **$19.50**

51955. A GUIDE TO THE SUCCESSFUL MANAGEMENT OF COMPUTER PROJECTS. *Hamish Donaldson.* **$24.95**

81845. SYSTEM/370 JOB CONTROL LANGUAGE. *Gary DeWard Brown.* Includes descriptions of hardware devices and access methods and reference to many useful JCL features. *Softcover.* **$13.50**

39995-2. COMPUTER NETWORKS AND THEIR PROTOCOLS. *D.W. Davies et al.* Latest technology in routing, pocket switching and flow control. *Counts as 2 of your 3 books.* **$43.95**

54455. IMS PROGRAMMING TECHNIQUES: A Guide to Using DL/1. *Kapp and Leben.* Specific techniques to write application programs in ANS COBOL, PL/1, or Assembler Language in an IMS DL/1 data base environment. **$17.95**

60385. MANAGING THE SYSTEMS DEVELOPMENT PROCESS. *Charles L. Biggs, et al.* **$24.95**

62620. MICROCOMPUTER HANDBOOK. *Charles J. Sippl.* Detailed reference to the technology and applications of microprocessors. **$19.95**

70110. PRINCIPLES OF DATA-BASE MANAGEMENT. *James Martin.* Covers data structures, security, file storage, real-time vs. non-real time, more. **$19.95**

70725. THE PROGRAMMER'S ANSI COBOL REFERENCE MANUAL. *Donald A. Sordillo.* **$22.50**

79150-2. SOFTWARE ENGINEERING. *Jensen & Tonies. Counts as 2 of your 3 books.* **$27.50**

79167. SOFTWARE RELIABILITY GUIDEBOOK. *Robert L. Glass.* Spells out all the technological and management techniques. **$18.95**

42053-2. DESIGNING A DISTRIBUTED PROCESSING SYSTEM. *Hamish Donaldson.* Casebook on how to manage projects successfully and a technical reference for the design problems that are peculiar to distributed processing. *Counts as 2 of your 3 books.* **$34.95**

Take any 3 books for only $1.00 each (values to $74.95)

If you will join now for a trial period and agree to take 3 more books—at handsome discounts—over the next 12 months

40135-2. A CONCISE DICTIONARY OF PHYSICS. *J. Thewlis.* More than 7,000 definitions. *Counts as 2 of your 3 books.* **$50.00**

42498. DOCUMENTATION STANDARDS AND PROCEDURES FOR ON-LINE SYSTEMS. *Edited by Martin L. Rubin.* A valuable guide to everything from user request documents to man/machine dialogue module for easier integration into existing company standards. **$21.95**

35450-2. AUTOMATIC DATA PROCESSING HANDBOOK. *The Diebold Group.* Auxiliary memory, peripherals, systems design, consultant services, software packages. *Counts as 2 of your 3 books.* **$38.95**

70230. PRINCIPLES OF SOFTWARE ENGINEERING AND DESIGN. *Marvin V. Zelkowitz, et al.* **$21.95**

40065-2. COMPUTER STORAGE SYSTEMS AND TECHNOLOGY. *Richard Matick. Counts as 2 of your 3 books.* **$29.95**

80150. STANDARDIZED DEVELOPMENT OF COMPUTER SOFTWARE. *Robert C. Tausworthe.* How to incorporate principles and techniques of structured programming in the development of large-scale systems. **$19.95**

80155. STANDARDIZED DEVELOPMENT OF COMPUTER SOFTWARE, PART II. *Robert C. Tausworthe.* A detailed set of rules for software implementation. Contains extensive appendices. **$21.95**

32485. ALGORITHMS + DATA STRUCTURES = PROGRAMS. *Niklaus Wirth.* A classic work on structured programming. **$20.95**

39890-2. COMPUTER DATA-BASE ORGANIZATION. *James Martin.* An invaluable planning tool having over 200 diagrams. *Counts as 2 of your 3 books.* **$26.50**

85725. UTILIZING SYSTEM 360/370 OS AND VS JOB CONTROL LANGUAGE AND UTILITY PROGRAMS. *Daniel H. Rindfleisch.* Practical guide to every feature of JCL and system utilities. **$21.95**

49600. THE FUTURE WITH MICROELECTRONICS. *Barron and Curnow.* Shows how future developments will transform every area of society from the factory to the home. **$17.50**

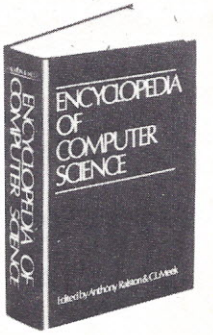
79155. SOFTWARE INTERPRETERS FOR MICROCOMPUTERS. *Thomas C. McIntire.* **$18.95**

50045-2. GAME THEORY: Mathematical Models of Conflict. *A. J. Jones.* Numerous diagrams, examples. *Counts as 2 of your 3 books.* **$45.00**

50551. GRANTS: How to Find Out About Them and What to Do Next. *Virginia P. White.* **$19.50**

39746-2. COMPANION TO CONCRETE MATHEMATICS. VOLUMES I AND II. *Z. A. Melzak.* Mathematical techniques and various applications. *The set counts as 2 of your 3 books.* **$46.45**

—EXTRAORDINARY— VALUE!



44900-3. THE ENCYCLOPEDIA OF COMPUTER SCIENCE. Monumental 1550-page volume offers accurate information on essential topics ranging from theory and applications to programming and Boolean algebra. *Counts as 3 of your 3 books.* **$60.00**

If the reply card has been removed, please write to **The Library of Computer and Information Sciences Dept 7-AG7, Riverside, N.J. 08075** to obtain membership information and application

Sturdy mechanical structure

Switchboard is nicely made with full solder masking and excellent silk-screened markings. The edge-board connector is gold plated for low noise, reliable operation, and the board is fully socketed, meaning that not only are there sockets provided for each of the memory devices, but all the ICs have sockets; a very nice touch. The layout is also well thought out (photo).

Switches SW3 through SW6 are located at the top edge along with the two input/output connectors (figure 1). These are the switches that are most likely to be changed during operation by the user, as they control the parallel, attention, and strobe ports as well as the memory addressing. The switches that control the two serial port UARTs are located close to these devices and the baud rate generator. Once the desired conditions for each serial port have been established, there isn't much need to change these switch settings unless one of the peripherals is changed, or replaced.

The I/O connectors are two 50-pin right-angle open style headers for the common flat cable. I would have preferred the use of the closed entry style connector for its resistance to damage, but this style is certainly acceptable. One other nice touch: the on-board crystal is held to the board by a piece of double-sided adhesive foam that provides shock resistance.

Professional circuit design

The board meets the proposed IEEE S-100 bus standard. All on-board supplies are generated by IC regulators with adequate capacitor filtering before and after the regulator to prevent noise and oscillations. The BR2941-L baud rate generator is used as the crystal oscillator as this is a function provided by that chip. The design of the TTY current loop and RS-232 interface is somewhat unusual (figure 2). The output function is performed by an operational amplifier driven from supply-rail to supply-rail to generate the required RS232C voltage swings.

Apparently Thinker Toys relies on the somewhat slower slew rate of the MC1458 operational amplifier to meet the maximum slew rate specifications ($\leqslant$30 v/usec.) of EIA RS232C. The typical slew rate of the MC1458 is 20 v/usec, but there is no specification for the maximum value, which could make this design a little borderline in high speed noisy situations.

The serial port receiver uses a 2-transistor circuit with essentially a grounded-base input configuration for the RS232C interface. With this configuration, it would appear that the space/mark input thresholds are only about +1/–1 volt rather than the +3/–3 volt specified by EIA RS232C. The noise immunity, as designed, should be sufficient for most normal applications and should not generally pose an operational problem. It would seem that the design could have been done just as easily with the standard RS232C interface integrated circuits—1488 and 1489. An opto-isolator could then have been used for the current loop to TTL conversion.

In actual use, the Switchboard is very easy to bring up and operate. The manual provides extensive coverage of board checkout and setup. Various shakedown programs are provided to test each type of port. □

Set fire to your paperwork.

Dispatch your paperwork with burning speed. With Spell Binder and a microcomputer you can set fire to just about any paperwork task in your office. Spell Binder is a software system that will drastically reduce the time you spend on paperwork, and ***increase*** your output.

WORD PROCESSING. Spell Binder is a word processing system you can learn to use in twenty minutes. But ease of use does not mean lack of powerful features. Compare Spell Binder's capabilities with any other word processing system. You'll be impressed. And Spell Binder doesn't stop with word processing.

MAILING LIST, MASS MAILING. Spell Binder features full mailing list and mailing label capability. And Spell Binder will merge your letter with your selected mailing list for personalized, tailored mass mailings. Sort and select by any key you want, including zip code.

CIRCLE INQUIRY NO. 8

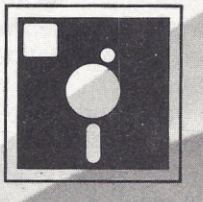
CUSTOM PROGRAMMING. Using Spell Binder's Macro Programming feature, you can shape your text to virtually any format, number the lines on legal text, create custom heads for running pagination, or do whatever you need for a fired up paperwork flow.

HOT, NEW, NOW. This new package is the best paperwork system you'll find on a microcomputer. It runs on computers with CP/M®, MP/M, OS-1®, and OASIS®. Spell Binder is available now. If it is not in your computer dealer's store for a demo, call us right away. Before your paperwork load turns into a real fire hazard.

OASIS is a registered trademark of Phase One Systems.
OS-1 is a registered trademark of Electrolabs.
CP/M, MP/M are registered trademarks of Digital Research.

CALIFORNIA PACIFIC COMPUTER CO.

7700 Edgewater Drive, Suite 747. Oakland, CA 94621.
(415) 569 9126



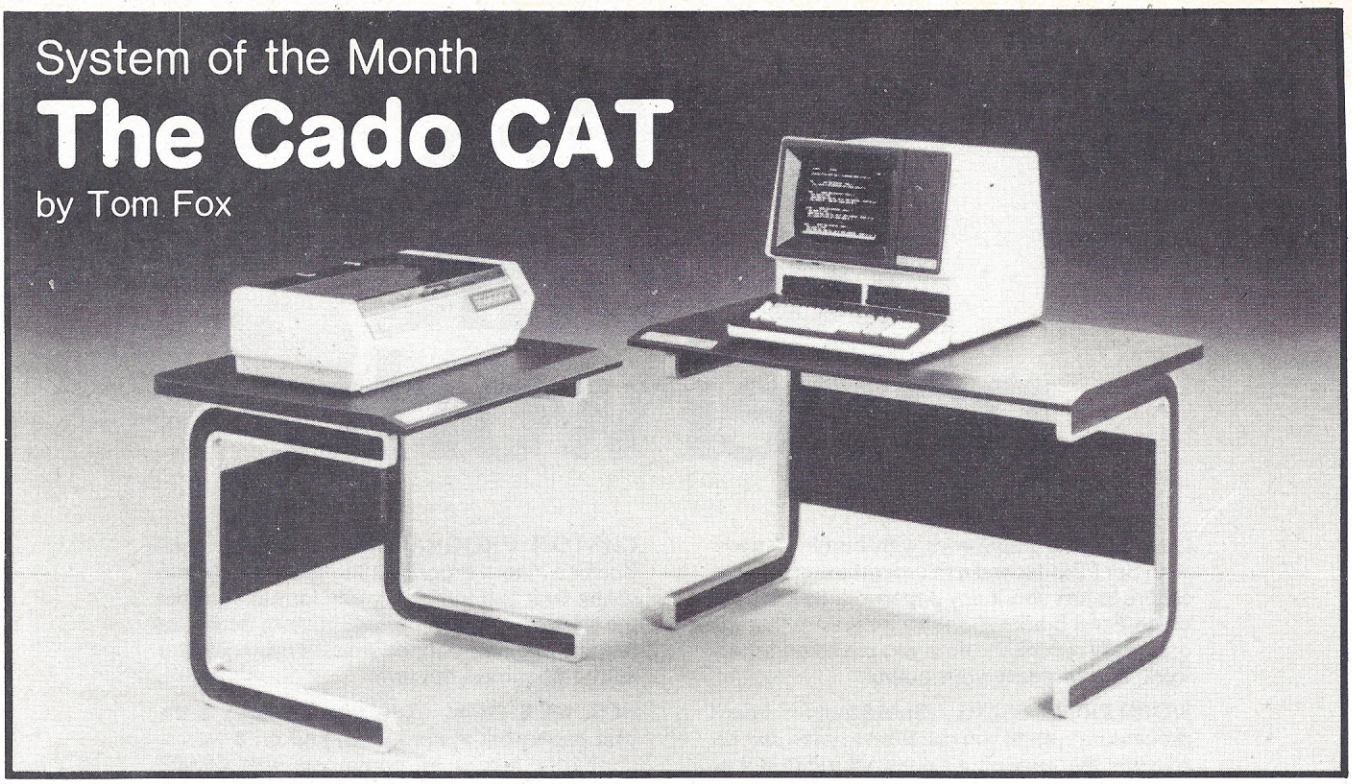
SPELL BINDER*

***A lot hotter than plain word processing.**

System of the Month

The Cado CAT

by Tom Fox



Usually a computer's name will tell something about what's inside the beast. Some monikers suggest the microprocessor chip utilized, the number of bits in each memory word or the size of the disk drive. In the case of the newest offering from Cado Systems Corp., Torrance, CA, the name stands for a family of software programs that runs on the machine, and for a marketing philosophy as well. CAT stands for computer aided tutor—you read right: a mechanical schoolmarm.

The CAT series 20/21 is the smallest sibling in Cado's family of computer products, including the 20/22, 20/24 and 20/28. The larger machines are two-, four- and eight-terminal systems, respectively. The CAT was created specifically for the first-time computer purchaser, and the concept is an important part of its marketing push.

Lesson number one

CAT computers are intended to be installed—unaided—by the nontechnical buyer. Only three units are plugged together, and an easy-to-follow written procedure is provided. Once this hurdle is conquered, the user only has to turn on the power and insert diskette #1.

Within minutes, the user is involved in a two-way educational dialog with the CAT. It's a programmed learning session that describes computer basics in a competent, entertaining manner and poses more advanced questions as the session proceeds. Correct answers are rewarded with new material; wrong ones with remedial explanations presented with typical computer patience.

Once you have mastered the first lesson, you can pick from eight others: accounting basics, accounts payable, accounts receivable, general ledger, payroll, inventory management, word processing and Just Ask. Each is backed up by an attractive three-color manual supplied in a loose-leaf binder. The lessons not only describe how to use the computer to—for example—handle a small company's payroll, but teach basic payroll concepts and lead you through the steps of converting to CAT's payroll system from a manual system or service bureau.

The tutorials were developed by Phoenix Performance Systems of St. Paul, MN. They were field tested, using inexperienced high school students, with gratifying results. A typical session takes three hours, but the rate of progression is up to the user. If outside business or brain saturation forces you to terminate a session, the CAT will return to where you left off when the machine is turned on again.

The computer-assisted path in making the owner self-reliant is central to the firm's sales philosophy. Cado intends to market through thousands of established office equipment stores, not computer outlets. Perhaps wisely, Cado will make no attempt to educate dealers in assembling or servicing computers, or in holding the hands of customers who can't make their computers cook. Faulty hardware will be replaced in toto at the dealer's complaint desk, rather than repaired at the end-user site. Cado intends that the CAT be sold and serviced like a desk calculator, and be just as easy to run.

All-in-one hardware configuration

The CAT is a member of the all-in-one desk-top computer packages, with the CPU, operator terminal and disk drives singly enclosed. The module appears to be a slightly enlarged, misshapen CRT terminal; a television-like display screen and entry keyboard comprise its only visible features.

The keyboard is detached from the main unit, connected by a generous length of durable-appearing coiled cable. This "floating" keyboard is more necessity than frill: the keyboard must be moved out of the way in order to insert diskettes into the floppy drives. The keyboard, incidentally, features many supplemental keys and special multi-color legends for the word processing program.

The two floppy disk drives are placed horizontally beneath the display screen. They are double-density,

giving approximately 1.2 mbytes (1,240,000 characters) of storage capacity on the two 8-inch diameter diskettes. The track-to-track access time of these units is 10 msec. The average time to read a piece of data on a diskette is said to be less than ½ second, taking into account all software and hardware delays. If more online storage is needed, the catalog lists a double-sided diskette drive option for an additional $1,000. The double-sided option not only doubles the storage capacity, but sports a track-to-track access time three times faster. There is currently no provision for adding additional diskette drives or for upgrading to a larger hard disk drive.

The display screen shines with a pleasing IBM-like green glow. Characters are fully formed and include many special symbols used to good advantage by word processing. Twenty-four lines of 80 characters each can be displayed at once, in addition to a bottom line in reverse video that displays the system status at all times. A nice touch.

The electronics are arranged neatly under the crisp black and white plastic housing in a way that shows careful thought to maintenance serviceability. The electronic functions of the CPU, memory, input/output and display screen control and refresh are distributed over three large circuit cards nestled in a four-card cage. These cards plug into a motherboard containing a bus structure of Cado's design. The internal arrangement and interconnection of parts are much neater and, we would guess, less trouble-prone than the similarly-proportioned Zenith Z89 or Pertec PCC 2000.

The chip is the 8-bit 8085 running a 3 MHz. It's surprising that it can only be had with 32K of RAM; competitors commonly feature 64K or more. The reason for this will be discussed along with the operating system.

The CAT's power supply regulates voltages by switching the current on and off at ultrasonic rates up to 40 KHz. The newer switching power supplies are several times more frugal in using energy resources and run much cooler, making life easier on all electronic parts.

At $13,900, the CAT comes complete with a 150 cps matrix printer. If typewriter-quality printouts are needed, add $2,000 to substitute the NEC Spinwriter. This is a slower 55-cps unit that uses a spinning thimble mechanism to print characters.

Distinctive desk-top software

The operating system is a stripped-down version of the multitasking software used in Cado's larger computers. Only two tasks are defined: the single user's terminal and the printer. This allows the printer and terminal to run at the same time, a real timesaver and definite plus in the field.

The operating system resides in 4K of ROM and a small amount of RAM. Each of the two users (terminal and printer) is allocated a mere 2K of memory space, none of which is used to store the running program. Applications programs created with Cadol are compiled into miniscule 256-byte chunks that are copied from the diskettes into a separate RAM area and run by the users as needed. This is called a "quasi-virtual memory" scheme adapted from much larger computer designs. The remaining memory is used to run assembly-language programs, utilities and subroutines. The whole arrangement optimizes the operation into a minimum of RAM space, at the cost of more frequent disk activity and (possibly) programmer convenience.

Basic-like language—but not quite

Cadol is the only high-level applications language available on the CAT. It is similar to Basic but simpler and lacking much of its sophistication. Cadol makes no pretense of being a scientific language. All numbers are 14-digit precision integers which eliminate rounding errors in financial calculations. (Double-precision routines were developed for use in Italy and Belgium, where businesses must keep track of liras and francs by the billions.) Each program only has 25 integer variables, prenamed N1 through N25. A similar limitation applies to strings, but three 80-byte variables, A, B and C, are allowed.

A special text editor is supplied for entering and debugging Cadol programs, and a compiler is included to convert a program into a semi-compiled run-time module. The compressed module is actually a series of one-byte subroutines that are interpreted by the operating system when the program runs. A 100-line Cadol program can typically be compiled into a 256-byte run-time module, the maximum size possible for a single program.

A good selection of 8085 assembly-language subroutines can be invoked by the Cadol programmer. These include a disk-based sort/merge routine, a Julian date conversion module and data compression call to optimize diskette storage space. These calls can provide up to a three-to-one compression for certain types of data files.

Additional software available

Just Ask is the catchy name for CAT's management inquiry system. Just Ask bears a more-than-coincidental resemblance to English, a popular database manager developed for Microdata Realty minicomputers. Just Ask can be configured by a nontechnical end user to handle a variety of data storage and report-generation tasks.

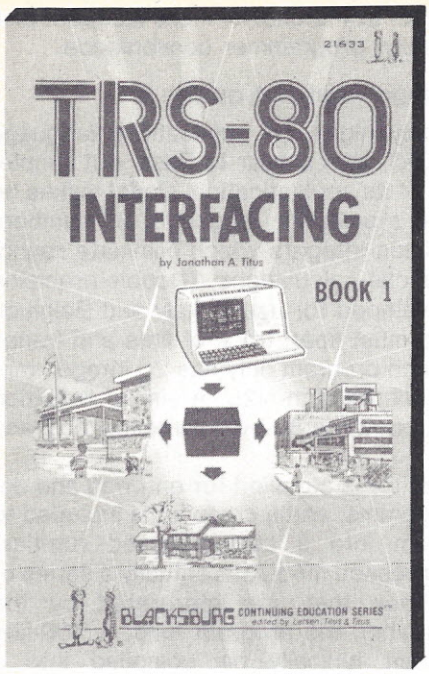
A final piece of software is one of the best: Word Processing. This program in 8085 assembly language features a screen-oriented input editor. Several of the keyboard keys are specially marked for functions such as underline, set margin, delete sentence, etc. The system contains more features than most will find use for, but it lacks little in utility. It works directly with disk files (to conserve RAM space) allowing very long documents to be handled. Many special routines are there to take full advantage of the Spinwriter's printing talents: margin justification by proportional justification, sub- and super-scripts, alternate type fonts, etc. It's a nice program, sure to cause a lot of dust to collect on old company typewriters.

Advantage: valuable software supplement

At $13,900, the CAT is priced within a few hundred dollars of IBM's 5120, another desk-top computer intended to be owner-installed by first-time users. The major difference is that the CAT includes a comprehensive software package that could add $10,000 to the 5120's price. Both manufacturers are targeting the over eight million under-$5 million small businesses in the country. □

Tom Fox can be reached at 17925-G Sky Park Circle, Irvine, CA 92714, (714) 957-9332.

NECESSITIES FOR THE COMPUTER USER



TRS-80 Interfacing Book I
By Jonathan A. Titus
Order No. 10058
192 pages $8.95

If you have a fairly good understanding of the commands in Level II BASIC, this book will appeal to you. The book introduces you to the signals available within the TRS-80 computer and shows you how they can be used to control external devices.

Computer Dictionary
By Donald D. Spencer
Order No. 19010
160 pages $5.95

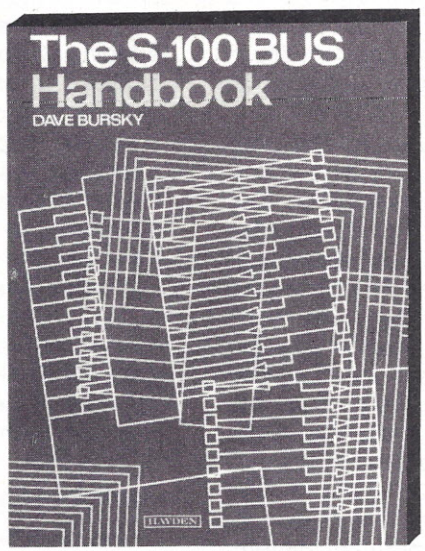
This current and comprehensive dictionary contains about 2,500 words, phrases, and acronyms used in connection with computers. The keynote of this book is clarity and precision without sacrifice of authority. All definitions are simple, and stand as independent units of explanation.

BASIC Programmer Primer
By Mitchell Waite and Michael Pardee
Order No. 10032
240 pages $8.95

This will serve as an invaluable tool to anyone who wants to learn BASIC, the most popular computer language of today. Seven chapters explain the fundamentals of BASIC, program control, organization and a game program, additional functions, and variations of BASIC.

Some Common BASIC Programs
By Lon Poole and Mary Borchers
Order No. 21001
200 pages $9.50

This book contains program listings and documentation for 76 short programs covering financial, mathematical, statistical and general interest applications. Every program has been tested; examples and sample program runs are published with the source listings for each program.



S-100 Bus Handbook
By David Bursky
Order No. 16033
280 pages $12.95

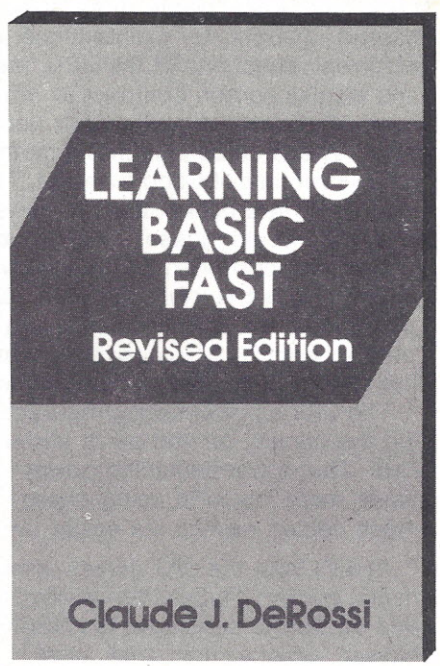
Computer fundamentals, basic electronics, and the parts of the computer are covered here. It explains all the operating details of commonly available S-100 systems. Schematic drawings of all the boards discussed in the manual are provided for reference. Each of the major system boards is discussed in complete detail as to how it operates and how it connects into the rest of the system.

Programming & Interfacing the 6502, With Experiments
By Marvin L. De Jong
Order No. 10059
448 pages $13.95

Filled with information and programs, this will be useful to anyone interested in using 6502-based microcomputer systems.

How to Program Microcomputers
By William Barden, Jr.
Order No. 10009
256 pages $8.95

This book explains assembly language programming of microcomputers. Basic concepts, including number systems and operations, basic computer operation, and computer codes are examined. Microcomputer architecture and operation are also discussed.



Learning BASIC Fast
By Claude Derossi
Order No. 20011
211 pages $11.95

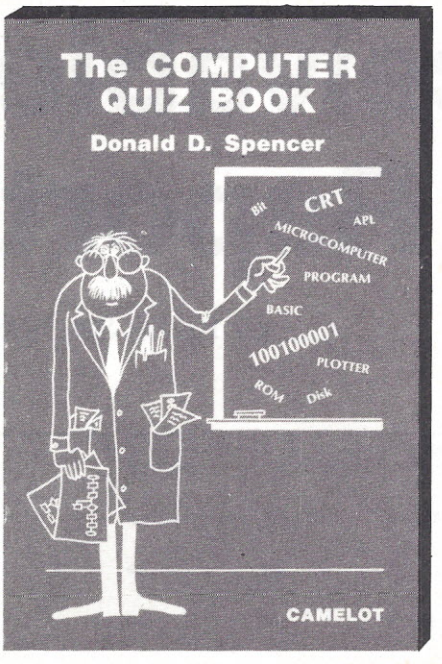
Provides simple, concise explanations of computer systems, flowcharting, and the important commands used in Basic. Requires no high-level math. Readers can begin programming almost immediately.

Peanut Butter and Jelly Guide to Computers
By Jerry Willis
Order No. 14022
225 pages $7.95

This delightful book begins with a series of case studies of different people who have units up and running. Readers are led through the essential information needed to become informed users of personal computers. The book is a combination consumers guide, tutorial and offers some thoughts on where you should buy your computer.

DATA DYNAMICS TECHNOLOGY

has the books to make it easy



The Computer Quiz Book

By Donald D. Spencer
Order No. 19009
128 pages $5.95

This book is written for readers who would like to test themselves on basic computer concepts. It may be used effectively by students, teachers, laymen, programmers, personal computer users, or anyone else interested in checking their knowledge of computer concepts.

From Dits to Bits

By Herman Lukoff
Order No. 14018
200 pages $12.95

Follow along as one of the pioneer developers of the computer recounts the intense atmosphere of discovery and frustration that pervaded the back room of Moore School at the University of Pennsylvania. The text is illustrated with numerous photographs that effectively recreate the atmosphere of the late 40's and early 50's. A glossary has been provided for the newcomer to the world of computers.

Build Your Own Working Robot

By David L. Heiserman
Order No. 13013
234 pages $5.95

Here are complete instructions, plans, schematics, logic circuits, and wiring diagrams for building Buster, the mechanical pet robot. He'll serve you coffee or bring you the morning paper. He'll forage for his own "food" and scream when he can't find it. His "curiosity" will get him into one plight after another, but Buster has the capacity to get himself out of trouble just as easily as he got into it!

How To Profit From Your Personal Computer: Professional, Business, and Home Applications

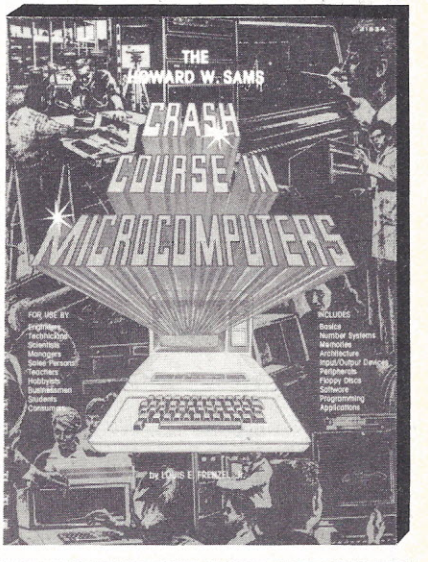
By T. G. Lewis
Order No. 16007
208 pages $9.65

Written for business people and computer hobbyists, this book shows you how to put your computer to work for you. It describes the uses of personal computers in common business applications, such as accounting, handling payrolls, managing inventory, sorting mailing lists, and many others. Many BASIC language programs are reproduced to illustrate the techniques for putting your computer to work for you.

The Howard W. Sams Crash Course in Microcomputers

By Louis E. Frenzel, Jr.
Order No. 10053
264 pages $17.50

Now you can stop struggling with conventional theory/reference-type computer books. The special "crash" course is organized into a series of 14 easy-to-learn units. This unique self-taught text gives you the most for your money and is a complete, fully-illustrated learning resource on the world of microcomputers.



9/80

DATA DYNAMICS TECHNOLOGY P.O. Box 1217, Cerritos, CA 90701

Name (Print) _____

Address _____

City _____ State _____ Zip _____

Please send me:

Order #	Qty	Price	Total	Order #	Qty	Price	Total

Shipping & Handling Charges $.75 ea. U.S., $1.50 ea. Foreign TOTAL ORDER $ _____

*TAX $ _____

SHIPPING & HANDLING $ _____

TOTAL ENCLOSED $ _____

☐ master charge ☐ AMERICAN EXPRESS ☐ VISA

_____ ☐ Check or M.O.(U.S. Funds drawn on U.S. bank)

Exp. Date _____ Signature _____

*California residents add 6% sales tax. Availability and prices quoted subject to change without notice.
Please allow six weeks for delivery. You may photocopy this page if you wish to keep your INTERFACE AGE intact.
Orders *cannot* be shipped unless accompanied by payment, including shipping & handling and tax where applicable.

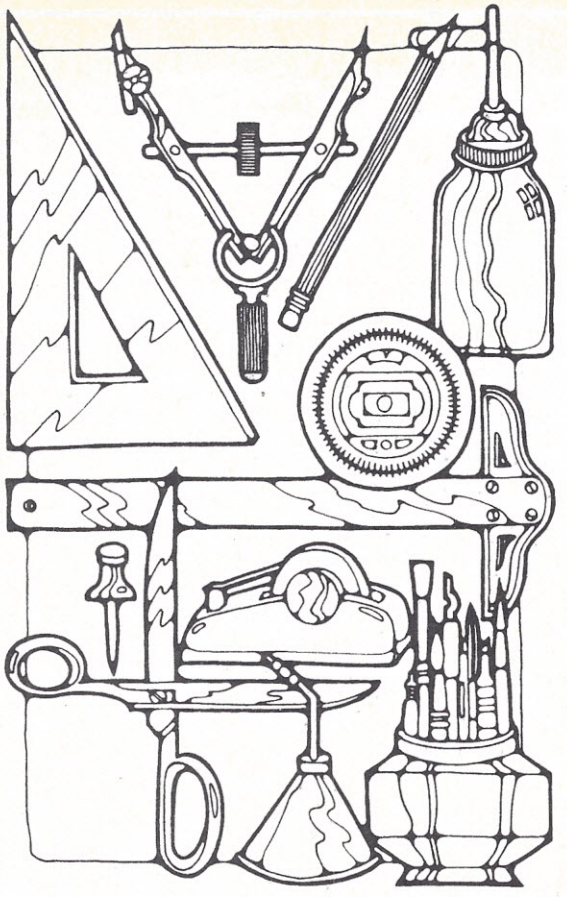
DATA DYNAMICS TECHNOLOGY, A Division of INTERFACE AGE Magazine (213) 926-9548

How to Design an Effective Accounts Receivable Report

by Kevin Stumpf

CUSTOMER NAME	#	BALANCES OWING TOTAL	CURRENT	30-DAYS	60-DAYS	90&OVER
ABC APPL & FURNITURE	51	1795.23	553.16	552.41	551.55	138.11
ABBOTT FURNITURE	52	-8845.50	-8845.50	.00	.00	.00
ABE'S FURNITURE LTD	53	7175.12	3948.42	3226.70	.00	.00
ABEL FURNITURE	54	-21930.50	-21930.50	.00	.00	.00
ACORES FURNITURE LTD	55	-6117.00	-6117.00	.00	.00	.00
ADELARD ALBERT; MEUBLES	56	-3909.00	-3909.00	.00	.00	.00
ALBERT'S DISCOUNT FURN	59	125.32	1.83	1.80	1.80	119.90
ALKEN FURNITURE LTD	62	497.87	7.25	7.14	7.14	476.33
ALLAIRE HOME FURNISHINGS	63	-5918.70	-5918.70	.00	.00	.00
ALLI'S HOUSE LTD	65	246.80	108.37	107.91	30.52	.00
APPLEBY FURNITURE	83	110.99	1.62	1.59	1.57	106.21
ARCTIC HOME FURNISHINGS	84	-3806.93	-3806.93	.00	.00	.00
ARLINGTON FURN CO LTD	85	-1324.86	-1324.86	.00	.00	.00
ARROW FURNITURE MART	86	3786.67	55.16	54.34	54.34	3622.83
ARTISTIC FURNITURE LTD	87	1520.78	271.61	267.90	256.89	724.38
ASSOFF'S FURNITURE	88	3000.44	593.05	584.54	567.54	1255.31
AUGUSTA FURN CO LTD	89	-2143.50	-2143.50	.00	.00	.00

Figure 1. Standard report format.



A popular computer application is printing customer account statements combined with accounts receivable reporting. These two functions are a good marriage because the data posted to the customer files to print statements is the same data summarized in the accounts receivable reports. A company's performance can be seen at a glance by reviewing accounts receivable reports showing overdue balances. The reports should be considered important, but too often are not designed for a variety of useful applications.

Let's investigate the ways such a report can be used and make a cursory study of the technical aspects of developing one. In a standard format (figure 1), the report has a title, a report date and a number printed on each page. The next detail is column headings. This report provides the reader with information concerning what customer (listed by name and number) owes money, how much, and a breakdown of that amount by equal periods of time. Notice the report is in alphabetical sequence and only those accounts with amounts owing are printed.

Collecting long overdue accounts

Go through the report line by line searching for accounts having amounts owing in the last column; in other words, those owing for a long time. Call these customers to inquire why the amount is still outstanding. This is the collection procedure most organizations follow. Why doesn't the report have a telephone number? Having to refer to the phone book or a file wastes time. Phone numbers should be stored in the customer master file and printed on the report.

CUSTOMER NAME/CONTACT	#	BALANCES OWING TOTAL	CURRENT	30-DAYS	60-DAYS	90&OVER	TELEPHONE #	REMARKS
ABC APPL & FURNITURE KEVIN	51 55	1795.23	553.16	552.41	551.55	138.11		
ABBOTT FURNITURE JANE	52 223	-8845.50	-8845.50	.00	.00	.00		
ABE'S FURNITURE LTD FRED	53	7175.12	3948.42	3226.70	.00	.00		
ABEL FURNITURE MURRY	54 444	-21930.50	-21930.50	.00	.00	.00		
ACORES FURNITURE LTD BARB	55 125	-6117.00	-6117.00	.00	.00	.00		
ADELARD ALBERT; MEUBLES TINA	56 34	-3909.00	-3909.00	.00	.00	.00		
ALBERT'S DISCOUNT FURN RYAN	59 5	125.32	1.83	1.80	1.80	119.90		
ALKEN FURNITURE LTD LYNN	62 45	497.87	7.25	7.14	7.14	476.33		
ALLAIRE HOME FURNISHINGS DORREEN	63 111	-5918.70	-5918.70	.00	.00	.00		
ALLI'S HOUSE LTD BOB	65 55	246.80	108.37	107.91	30.52	.00	519-323-1721	
APPLEBY FURNITURE JOE	83 6	110.99	1.62	1.59	1.57	106.21	416-681-0100	
ARCTIC HOME FURNISHINGS	84	-3806.93	-3806.93	.00	.00	.00		

Figure 2. Listing contact person.

Another worthy inclusion is space to write important remarks during the phone call. For instance, if collection calls are made on a weekly basis and a customer says he will remit in two weeks, you would be wasting time by calling every week, and might also create some antagonism.

If enough diskette space is available, it might be possible to include the name of a contact person and his extension number. Calling the same person regularly and knowing his name develops a much needed rapport (figure 2).

In some cases, the collection officer is instructed to call only customers with excessive overdue amounts. Why not produce a report with only the accounts meeting these qualifications? It saves paper and restricts confidential information from floating around the organization.

Collection of overdue accounts is not just the simple process of seeing an amount of money owing on a report and quickly calling the cusomer to demand the money. Policies such as credit limits and purchase/sales histories will affect the decision of when and how to approach the customer. A report as shown in figure 3 might then be useful.

Include sales analysis

Including purchase/sales histories in the report introduces a new application. Sales analysis is extremely important to any sales oriented organization. The report shown in figure 4 was designed for this purpose. The report is printed by sales territory for each sales person in account number sequence. The reason the sequence is not alphabetical is primarily economic. Sorting takes time, especially on microcomputers. And since this report is not intended to be used by office staff making telephone collections, no benefit is gained from alphabetic groupings. Not only do accounts owing money appear on the report but accounts with no sales this month also appear. Printing the accounts showing no purchases/sales is intended to be an inspiration to the salesperson to make that zero go away next month.

A few final hints

What's in a title? Everything. The examples in the article use conservative, familiar report titles, but if everyone in the organization calls the manually-prepared accounts receivable list the "scandal sheet," then why not program it that way.

In every case, print the customer name first if the sequence is alphabetical. Reverse the order if the sequence is by account number. This picks up the needed detail faster. Always print both the customer name and number. Computers are built to work with numbers; people are not. When a customer complains that no money is owing, the account number is the vital reference needed to sift through the input audit lists to find the error. On the other hand, when a person uses the report, it is much nicer to know the A.B. Corporation owes money, not just account 137.

CUSTOMER NAME/CONTACT	#	S C	CREDIT LIMIT	YTD PUR	BALANCES OWING TOTAL	CURRENT	30-DAYS	60-DAYS	90&OVER	TELEPHONE #	REMARKS
ABC APPL & FURNITURE KEVIN	51 55	6	5000	542	1795.23	553.16	552.41	551.55	138.11		
ABBOTT FURNITURE JAN223	52 223	1	1000	0	-8845.50	-8845.50	.00	.00	.00		
ABE'S FURNITURE LTD FRED	53	12	1000	1326	7175.12	3948.42	3226.70	.00	.00		
ABEL FURNITURE MURRY	54 444	5	700	0	-21930.50	-21930.50	.00	.00	.00		
ACORES FURNITURE LTD BARB	55 125	9	700	0	-6117.00	-6117.00	.00	.00	.00		
ADELARD ALBERT; MEUBLES TINA	56 34	07	700	0	-3909.00	-3909.00	.00	.00	.00		
ALBERT'S DISCOUNT FURN RYAN	59 5	9	1000	0	125.32	1.83	1.80	1.80	119.90		
ALKEN FURNITURE LTD LYNN	62 45	9	700	0	497.87	7.25	7.14	7.14	476.33		
ALLAIRE HOME FURNISHINGS DORREEN	63 111	3	2000	0	-5918.70	-5918.70	.00	.00	.00		
ALLI'S HOUSE LTD BOB	65 55	1	0	107	246.80	108.37	107.91	30.52	.00	519-323-1721	
APPLEBY FURNITURE JOE	83 6	4	1000	0	110.99	1.62	1.59	1.57	106.21	416-681-0100	
ARCTIC HOME FURNISHINGS	84	3	1000	0	-3806.93	-3806.93	.00	.00	.00		

Figure 3. Purchase/sales histories.

The program listing begins with a layout of a record from the customer account master file highlighting pertinent fields. The file handling technique uses a single character string to store all fields contiguously. This approach is used to optimize disk space usage and input/output operations.

Sometimes telephone numbers are difficult to obtain. In some cases, useless dashes get printed. These are printed by the program to separate the area code from the exchange from the subscriber's number portions of a phone number. In this program, lines 5050 to 5052, the presence and length of a telephone number are determined and the printing is done accordingly. □

AGED BALANCE REPORT BY TERRITORY- 1 SALPERSON - SMITH AGENCY

CUSTOMER NAME/CITY/#	TOTAL	CURRENT	OVERDUE 30-DAYS	OVERDUE 60-DAYS	OVERDUE 90&OVER
HOME HARDWARE STORES LTD ST JACOBS ON	-9355.96 003	-9355.96	.00	.00	.00
~~ABBOTT FURNITURE~~ GUELPH ON	~~8845.50~~ 52	~~8845.50~~	~~.00~~	~~.00~~	.00
ALLI'S HOUSE LTD MOUNT FOREST ON	246.80 65	108.37	107.91	30.52	.00
BALL FURNITURE ST. MARY'S ON	.00 94	.00	.00	.00	.00
BALL & MUTCH FURNITURE CLINTON ON	.00 95	.00	.00	.00	.00
BEATTIE FURNITURE CLINTON ON	-7.47 104	-7.47	.00	.00	.00
BLACKSTONE FURNITURE GODERICH ON	.00 121	.00	.00	.00	.00
BONTHRON FURNITURE HENSALL ON	.00 126	.00	.00	.00	.00
BOX FURNITURE STORE SEAFORTH ON	.00 131	.00	.00	.00	.00
BOX & SON; M. PARKHILL ON	3567.98 132	734.69	724.14	703.05	1406.10
BURNSIDE HOME FURNISHING MARKSDALE ON	.00 142	.00	.00	.00	.00
BUTCHER FURNITURE; J. D. HILLSBURGH ON	-130.80 144	-130.80	.00	.00	.00
CURRIE FURNITURE; JAMES L CHATSWORTH ON	-1498.19 183	-1498.19	.00	.00	.00
DAVID'S DECORS LTD TAVISTOCK, ONT	248.89 188	33.26	32.78	32.31	150.53
DIEGEL'S DEPT. STORE MITCHELL ON	.00 201	.00	.00	.00	.00
DOUGLAS FURNITURE LTD	255.51	3.72	3.67	3.61	

Figure 4. Sales analysis report.

Program follows

Start Computing For Just $129.95 With An 8085-Based Professional Computer Kit—

Explorer/85

100% compatible with all 8080A and 8085 software & development tools!

No matter what your future computing plans may be, Level "A"—at $129.95—is your starting point.

Starting at just $129.95 for a Level "A" operating system, you can now build the exact computer you want. Explorer/85 can be your beginner's system, OEM controller, or IBM-formatted 8" disk small business system...yet you're never forced to spend a penny for a component or feature you don't want and you can expand in small, affordable steps!

Now, for just $129.95, you can own the first level of a fully expandable computer with professional capabilities—a computer which features the advanced Intel 8085 cpu, *thereby giving you immediate access to all software and development tools that exist for both the 8085 and its 8080A predecessor (they are 100% software compatible)*—a computer which features onboard S-100 bus expansion—plus instant conversion to mass storage disk memory with either 5-1/4" diskettes or standard IBM-formatted 8" disks.

For just $129.95 (plus the cost of a power supply, keyboard/terminal and RF modulator, if you don't have them already), Explorer/85 lets you begin *computing* on a significant level...*applying* the principles discussed in leading computer magazines...*developing* "state of the art" computer solutions for both the industrial and leisure environment.

Level "A" Specifications

Explorer/85's Level "A" system features the advanced Intel 8085 cpu, an 8355 ROM with 2k deluxe monitor/operating system, and an 8155 ROM-I/O—all on a single motherboard with room for RAM/ROM/PROM/EPROM and S-100 expansion, *plus* generous prototyping space.

(Level "A" makes a perfect OEM controller for industrial applications and is available in a special Hex Version which can be programmed using the Netronics Hex Keypad/Display.)

Level "A" at $129.95 is a complete operating system, perfect for beginners, hobbiests, or industrial controller use.

PC Board: glass epoxy, plated through holes with solder mask • **I/O:** provisions for 25-pin (DB25) connector for terminal serial I/O, which can also support a paper tape reader ...provision for 24-pin DIP socket for hex keyboard/display...cassette tape recorder input...cassette tape recorder output...cassette tape control output...speaker output... LED output indicator on SOD (serial output) line...printer interface (less drivers)...total of four 8-bit plus one 6-bit I/O ports • **Crystal Frequency:** 6.144 MHz • **Control Switches:** reset and user (RST 7.5) interrupt...additional provisions for RST 5.5, 6.5 and TRAP interrupts onboard • **Counter/Timer:** programmable, 14-bit binary • **System RAM:** 256 bytes located at F8ØØ, ideal for smaller systems and for use as an isolated stack area in expanded systems...RAM expandable to 64k via S-100 bus or 4K on motherboard.

System Monitor (Terminal Version): 2k bytes of deluxe system monitor ROM located at FØØØ leaving ØØØØ free for user RAM/ROM. Features include tape load with labeling ...tape dump with labeling...examine/change contents of memory ...insert data...warm start...examine and change all registers...single step with register display at each break point, a debugging/training feature...go to execution address... move blocks of memory from one location to another...fill blocks of memory with a constant...display blocks of memory ...automatic baud rate selection...variable display line length control (1-255 characters/line)...channelized I/O monitor routine with 8-bit parallel output for high speed printer... serial console in and console out channel so that monitor can communicate with I/O ports.

System Monitor (Hex Version): Tape load with labeling... tape dump with labeling...examine/change contents of memory...insert data...warm start...examine and change all registers...single step with register display at each break point ...go to execution address. Level "A" in the *Hex Version* makes a perfect controller for industrial applications and can be programmed using the Netronics Hex Keypad/Display.

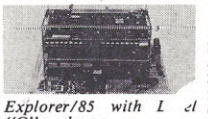
Hex Keypad/Display.

Hex Keypad/Display Specifications

Calculator type keypad with 24 system defined and 16 user defined keys. 6 digit calculator type display which displays full address plus data as well as register and status information.

Level "B" Specifications

Level"B" provides the S-100 signals plus buffers/drivers to support up to six S-100 bus boards and includes: address decoding for onboard 4k RAM expansion select-able in 4k blocks...address decoding for onboard 8k EPROM expansion selectable in 8k blocks...address and data bus drivers for onboard expansion...wait state generator (jumper selectable), to allow the use of slower memories...two separate 5 volt regulators.



Explorer/85 with Level "C" card cage.

Level "C" Specifications

Level "C" expands Explorer's motherboard with a card cage, allowing you to plug up to six S-100 cards directly into the motherboard. Both cage and cards are neatly contained inside Explorer's deluxe steel cabinet. Level "C" includes a sheet metal superstructure, a 5-card gold plated S-100 extension PC board which plugs into the motherboard. Just add required number of S-100 connectors

Level "D" Specifications

Level "D" provides 4k or RAM, power supply regulation, filtering decoupling components and sockets to expand your Explorer/85 memory to 4k (plus the original 256 bytes located in the 8155A). The static RAM can be located anywhere from ØØØØ to EFFF in 4k blocks.

Level "E" Specifications

Level "E" adds sockets for 8k of EPROM to use the popular Intel 2716 or the TI 2516. It includes all sockets, power supply regulator, heat sink, filtering and decoupling components. Sockets may also be used for soon to be available RAM IC's (allowing for up to 12k of onboard RAM).

Order A Coordinated Explorer/85 Applications Pak!

Experimenter's Pak (SAVE $12.50)—Buy Level "A" and Hex Keypad/Display for **$199.90** and get FREE Intel 8085 user's manual plus FREE postage & handling!

Student Pak (SAVE $24.45)—Buy Level "A," ASCII Keyboard/Computer Terminal, and Power Supply for **$319.85** and get FREE RF Modulator plus FREE Intel 8085 user's manual plus FREE postage & handling!

Engineering Pak (SAVE $41.00)—Buy Levels "A," "B," "C," "D," and "E" with Power Supply, ASCII Keyboard/Computer Terminal, and six S-100 Bus Connectors for **$514.75** and get 10 FREE computer grade cassette tapes plus FREE 8085 user's manual plus FREE postage & handling!

Business Pak (SAVE $89.95)—Buy Explorer/85 Levels "A," "B," and "C" (with cabinet), Power Supply, ASCII Keyboard/Computer Terminal (with cabinet), 16k RAM, 12" Video Monitor, North Star 5-1/4" Disk Drive (includes North Star BASIC) with power supply and cabinet, all for just **$1599.40** and get 10 FREE 5-1/4" minidiskettes ($49.95 value) plus FREE 8085 user's manual plus FREE postage & handling!

Continental U.S.A. Credit Card Buyers Outside Connecticut

CALL TOLL FREE 800-243-7428

To Order From Connecticut Or For Technical Assistance, Etc. **Call (203) 354-9375**

Netronics R&D Ltd., Dept. RE 10
333 Litchfield Road, New Milford, CT 06776

Please send the items checked below—

☐ **Explorer/85 Level "A" Kit** (ASCII Version), **$129.95** plus $3 p&h.

☐ **Explorer/85 Level "A" Kit** (Hex Version), **$129.95** plus $3 p&h.

☐ **8k Microsoft BASIC** on cassette tape, **$64.95** postpaid.

☐ **8k Microsoft BASIC in ROM Kit** (requires Levels "B," "D," and "E"), **$99.95** plus $2 p&h.

☐ **Level "B" (S-100) Kit, $49.95** plus $2 p&h.

☐ **Level "C" (S-100 6-card expander) Kit, $39.95** plus $2 p&h.

☐ **Level "D" (4k RAM) Kit, $69.95** plus $2 p&h.

☐ **Level "E" (EPROM/ROM) Kit, $5.95** plus 50¢ p&h.

☐ **Deluxe Steel Cabinet** for Explorer/85, **$49.95** plus $3 p&h.

☐ **ASCII Keyboard/Computer Terminal Kit** (features a full 128 character set, upper & lower case, full cursor control, 75 ohm video output convertible to baudot output, selectable baud rate, RS232-C or 20 ma. I/O, 32 or 64 character by 16 line formats, and can be used with either a CRT monitor or a TV set (if you have an RF modulator), **$149.95** plus $2.50 p&h.

☐ **Hex Keypad/Display Kit, $69.95** plus $2 p&h.

☐ **Deluxe Steel Cabinet** for ASCII Keyboard/Terminal, **$19.95** plus $2.50 p&h.

☐ **Power Supply Kit** (±8V @ 5 amps) in deluxe steel cabinet, **$39.95** plus $2 p&h.

☐ **Gold Plated S-100 Bus Connectors, $4.85** each, postpaid.

☐ **RF Modulator Kit** (allows you to use your TV set as a monitor), **$8.95** postpaid.

☐ **16k RAM Kit** (S-100 Board expands to 64k), **$199.95** plus $2 p&h.

☐ **32k RAM Kit, $329.95** plus $2 p&h.

☐ **48K RAM Kit, $459.95** plus $2 p&h.

☐ **64k RAM Kit, $589.95** plus $2 p&h.

☐ **16k RAM Expansion Kit** (to expand any of the above up to 64k), **$139.95** plus $2 p&h each.

☐ **Intel 8085 cpu User's Manual, $7.50** postpaid.

☐ **Special Computer Grade Cassette Tapes, $1.90** each or **3 for $5,** postpaid.

☐ **12" Video Monitor** (10 MHz bandwidth), **$139.95** plus $5 p&h.

☐ **North Star Double Density Floppy Disk Kit** (One Drive) for Explorer/85 (includes 3 drive S-100 controller, DOS, and extended BASIC with personalized disk operating system—just plug it in and you're up and running!), **$699.95** plus $5 p&h.

☐ **Power Supply Kit** for North Star Disk Drive, **$39.95** plus $2 p&h.

☐ **Deluxe Case** for North Star Disk Drive, **$39.95** plus $2 p&h.

☐ **Experimenter's Pak** (see above), **$199.90** postpaid.

☐ **Student Pak** (see above), **$319.85** postpaid.

☐ **Engineering Pak** (see above), **$514.75** postpaid.

☐ **Business Pak** (see above), **$1599.40** postpaid.

Total Enclosed $ _____
(Conn. res. add sales tax) By—
☐ Personal Check ☐ M.O./Cashier's Check ☐ Visa ☐ Master Charge (Bank # ____)

Acct. # _____

Signature _____ Exp. Date ____

Print Name _____

Address _____

City _____

State _____ Zip _____

☐ **Send Me Information**

By Netronics

ASCII/BAUDOT, STAND ALONE

Computer Terminal

COMPLETE FOR ONLY... $149.95

The Netronics ASCII/BAUDOT Computer Terminal Kit is a microprocessor-controlled, stand alone keyboard/terminal requring no computer memory or software. It allows the use of either a 64 or 32 character by 16 line professional display format with selectable baud rate, RS232-C or 20 ma. output, full cursor control and 75 ohm composite video output.

The keyboard follows the standard typewriter configuration and generates the entire 128 character ASCII upper/lower case set with 96 printable characters. Features include onboard regulators, selectable parity, shift lock key, alpha lock jumper, a drive capability of one TTY load, and the ability to mate directly with almost any computer, including the new Explorer/85 and ELF products by Netronics.

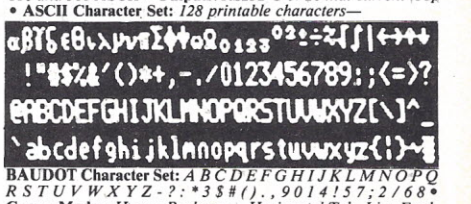
The Computer Terminal requires no I/O mapping and includes 1k of memory, character generator, 2 key rollover, processor controlled cursor control, parallel ASCII/BAUDOT to serial conversion and serial to video processing—fully crystal controlled for superb accuracy. PC boards are the highest quality glass epoxy for the ultimate in reliability and long life.

VIDEO DISPLAY SPECIFICATIONS

The heart of the Netronics Computer Terminal is the microprocessor-controlled Netronics Video Display Board (VID) which allows the terminal to utilize either a parallel ASCII or BAUDOT signal source. The VID converts the parallel data to serial data which is then formatted to either RS232-C or 20 ma. current loop output, which can be connected to the serial I/O on your computer or other interface, i.e., Modem.

When connected to a computer, the computer must echo the character received. This data is received by the VID which processes the information, converting to data to video suitable to be displayed on a TV set (using an RF modulator) or on a video monitor. The VID generates the cursor, horizontal and vertical sync pulses and performs the housekeeping relative to which character and where it is to be displayed on the screen.

Video Output: *1.5 P/P into 75 ohm (EIA RS-170)* • **Baud Rate:** *110 and 300 ASCII* • **Outputs:** *RS232-C or 20 ma. current loop* • **ASCII Character Set:** *128 printable characters—*



BAUDOT Character Set: *A B C D E F G H I J K L M N O P Q R S T U V W X Y Z - ? : * 3 $ # ( ) . , 9 0 1 4 ! 5 7 ; 2 / 6 8* • **Cursor Modes:** *Home, Backspace, Horizontal Tab, Line Feed, Vertical Tab, Carriage Return. Two special cursor sequences are provided for absolute and relative X-Y cursor addressing* • **Cursor Control:** *Erase, End of Line, Erase of Screen, Form Feed, Delete* • **Monitor Operation:** *50 or 60Hz (jumper selectable.*

Continental U.S.A. Credit Card Buyers Outside Connecticut

CALL TOLL FREE 800-243-7428

To Order From Connecticut Or For Technical Assistance, Etc. **Call (203) 354-9375**

Netronics R&D Ltd., Dept. PE-9
333 Litchfield Road, New Milford, CT 06776

Please send the items checked below—

☐ **Netronics Stand Alone ASCII Keyboard/Computer Terminal Kit, $149.95** plus $3.00 postage & handling.

☐ **Deluxe Steel Cabinet** for Netronics Keyboard/Terminal In Blue/Black Finish, **$19.95** plus $2.50 postage and handling.

☐ **Video Display Board Kit** alone (less keyboard), **$89.95** plus $3 postage & handling.

☐ **12" Video Monitor** (10 MHz bandwidth) fully assembled and tested, **$139.95** plus $5 postage and handling.

☐ **RF Modulator Kit** (to use your TV set for a monitor), **$8.95** postpaid.

☐ **5 amp Power Supply Kit** In Deluxe Steel Cabinet (±8VDC @ 5 amps, plus 6-8 VAC), **$39.95** plus $2 postage & handling.

Total Enclosed (Conn. res. add sales tax) $_____
By—
☐ Personal Check ☐ Cashiers Check/Money Order
☐ **Visa** ☐ **Master Charge** (Bank # ____)

Acct. # _____

Signature _____ Exp. Date _____

Print Name _____

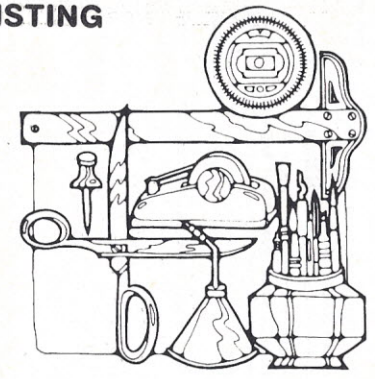
Address _____

City _____

State _____ Zip _____

☐ **Send Me More Information**

PROGRAM LISTING



```
1 REM SAMPLE PROGRAM WRITTEN IN NORTHSTAR BASIC BY
2 REM KEVIN STUMPF - COMPUTING ELEMENTS DEVELOPMENT
3 REM
10 DIM R$(160),Z$(25)
11 REM ***********************************************
12 REM R$ - ACCOUNT MASTER FILE RECORD LAYOUT
13 REM            ACCOUNT #          1 -   4
14 REM            ACCOUNT NAME       5 -  25
15 REM            MAILING INFO      30 -  50
16 REM            TOTAL OWING       81 -   9
17 REM            TELEPHONE #       90 -  10
18 REM            CURRENT BALANCE  100 -   8
19 REM            30-DAY BALANCE   108 -   8
20 REM            60-DAY BALANCE   116 -   8
21 REM            90 DAYS & OVER   124 -   8
22 REM            MISC INFO        132 -  15
23 REM            CREDIT LIMIT     147 -   1
24 REM            SPARE            148 -   1
25 REM            YTD PURCHASES    149 -   6
26 REM            SPARE            155 -   6
30 REM ***********************************************
35 REM OPERATOR INSTRUCTIONS
40 !CHR$(27),CHR$(42)\!"PAGEAR - PRINT AGED ACCOUNT BALANCES REPORT"\!\!
45 !"PLACE ACCOUNT MASTER DISKETTE IN RIGHT SLOT THEN 'RETURN'."
50 X$=INCHAR$(0)\Z=FILE("FACOUNTM,2")\IFZ<>3THEN45
55 !\INPUT"ENTER REPORT DATE (DDMMMYY) - ",D$\IFLEN(D$)<>7THEN55
60 !\!"MAKE SURE WIDE PAPER IN PRINTER LINED UP TO NEW PAGE & THE SELECT"
65 !"LIGHT IS ON THEN 'RETURN'."\X$=INCHAR$(0)
70 REM
71 REM INITIALIZE COUNTERS & ACCUMMULATORS
72 REM
75 T1=0\T2=0\T3=0\T4=0\T5=0\L=0\P=1
80 REM
81 REM USE A MULTI-PASS ALPHANUMERIC SORT ON FIRST CHARACTER OF NAME
82 REM
100 FORI=48TO57\GOSUB115\NEXTI
101 FORI=65TO90\GOSUB115\NEXTI
105 GOTO187
115         FOR K = 1 TO 1086
120             READ #1 %(K-1)*162,R$
125             IF R$(1,4)="0000" THEN 140
129             IFASC(R$(5,5))<>ITHEN140
130             IFABS(VAL(R$(81,89)))=0THEN140
135             GOSUB5000
140         NEXT K
145 RETURN
160 REM
161 REM PRINT REPORT TOTALS
162 REM
187 GOSUB6000
190 CLOSE#1
200 STOP
4999 REM PRINT LINE
5000 IFL<>0ANDL<54THEN5020
5005 IFL=0THEN5015
5010 FORJ=1TO6\!#7,CHR$(13)\NEXTJ
5015 GOSUB8000
5020 !#7,R$(5,29),TAB(25),%4I,VAL(R$(1,4))," ",R$(79,80),
5025 M=VAL(R$(147,147))\IFM=0ORM>6THEN5040
5027 REM DETERMINE THE CREDIT LIMIT IN DOLLARS GIVEN CODE
5030 ONMGOTO5031,5032,5033,5034,5035,5036
5031 C=500\GOTO5037
5032 C=700\GOTO5037
5033 C=1000\GOTO5037
5034 C=2000\GOTO5037
5035 C=5000\GOTO5037
5036 C=10000
5037 GOTO5045
5040 C=0
5045 !#7,TAB(33),%5I,C," ",%6I,VAL(R$(149,154)),
5047 !#7,%10F2,VAL(R$(81,89))," ",%10F2,VAL(R$(100,107)),
5048 !#7," ",%10F2,VAL(R$(108,115))," ",%10F2,VAL(R$(116,123)),
5049 !#7," ",%10F2,VAL(R$(124,131)),
5050 IFVAL(R$(90,99))=0THEN5053ELSEIFR$(97,99)="   "THEN5052
5051 !#7," ",R$(90,92),"-",R$(93,95),"=",R$(96,99)\GOTO5053
5052 !#7,"       ",R$(90,92),"-",R$(93,95)\GOTO5053
5053 !#7,TAB(120),". . . . . "
5055 !#7,CHR$(13)\L=L+2
5057 REM ACCUMULATE REPORT TOTALS
5060 T1=T1+VAL(R$(81,89))
5065 T2=T2+VAL(R$(100,107))
5070 T3=T3+VAL(R$(108,115))
5075 T4=T4+VAL(R$(116,123))
5080 T5=T5+VAL(R$(124,131))
5085 RETURN
6000 REM
6010 PRINT#7,CHR$(13)
6015 PRINT#7,"******* TOTALS",
6020 PRINT#7,TAB(44),%$C12F2,T1,TAB(56),%12F2,T2,TAB(68),%12F2,T3,
6025 PRINT#7,TAB(80),%12F2,T4,TAB(92),%12F2,T5
6035 RETURN
8000 !#7,CHR$(14),D$,TAB(23),"MAGNUSSEN FURNITURE      PAGE - ",%3I,P
8015 !#7,TAB(55),"A/R AGED BALANCES"
8020 !#7,CHR$(13)
8025 !#7,"---------- CUSTOMER ---------- S CREDIT   YTD ----------",
8030 !#7,"---------- BALANCES OWING ----------   TELEPHONE"
8035 !#7,TAB(9),"NAME",TAB(27),"#   C  LIMIT   PUR      TOTAL",
8040 !#7,TAB(60),"CURRENT     30-DAYS    60-DAYS      90&OVER",
8043 !#7,"          #     REMARKS"
8045 !#7,CHR$(13)\L=0
8050 P=P+1
8055 RETURN
9999 END
```

FINANCIAL ACCOUNTING SOFTWARE

A generalized, interactive bookkeeping and accounting system created by our staff of CPA's for our accounting practice. The system includes all of the accounting journals, ledgers, and reports necessary for a complete accounting package (i.e. general ledger, payroll, receivables, financial statements, plus a few things not found elsewhere.) This is a tried and true system with over three years of six and seven day per week usage.

Source code to the PJA Accounting System with documentation is available for $500. An Alpha Micro computer with either floppy or hard disk is required. Dealerships are available.

TAX PREPARATION SOFTWARE

An on-line, interactive, individual income tax preparation system created by our staff of CPA's for our tax practice. The system prepares form 1040 plus 24 related forms and schedules found in typical returns. This system allows the tax professional to greatly increase productivity, accuracy, and profitability. This is a tried and true system with two tax seasons behind it.

Source code to the PJA Tax System with documentation is available for $800. An Alpha Micro computer with either floppy or hard disk is required. Dealerships are available.

Payne, Jackson and Associates

CERTIFIED PUBLIC ACCOUNTANTS

611 West Ninth Avenue
Anchorage, Alaska 99501
(907) 272-7261 or 279-2351

CIRCLE INQUIRY NO. 50

CIRCLE INQUIRY NO. 51

Free Literature

- **M2114 Static RAM.** A six-page data sheet listing the AC and DC operating characteristics, waveforms, and ordering information for the 300 and 450 RAMs. GTE Communications Dept., 1 Stamford Forum, Stamford CT 06904. CIRCLE INQUIRY NO. 200

- **Development System/Control Computer.** A 40-page catalog with specifications on the Sprint 68 single board system, plus alternative software developments, educational services, and cross software products. Wintek, 1801 S. St., Lafayette IN 47904. CIRCLE INQUIRY NO. 201

- **Ribbons and Printwheels.** An eight-page catalog for Diablo/Xerox, Qume, and N.E.C. equipment, including cartridge ribbons, printwheels, and data recording devices. Swallow Industries, Inc., 3002 Hadley Rd., South Plainfield NJ 07080. CIRCLE INQUIRY NO. 202

- **NTC Thermistors.** A two-page application note on negative temperature coefficient thermistors including design parameters for bead, chip, wafer, interchangeable wafer and flake configurations. Bulletin ICN 7907. Piezo Products Div., Gulton Industries, 212 Durham Ave., Metuchen NJ 08840. CIRCLE INQUIRY NO. 203

- **Build a Microprocessor.** An 80-page product guide, MPG-190C, describes the complete line of ICs, support systems, and accessories that constitute the RCA 1800 Cosmac microprocessor family. RCA Solid State Div., Box 3200, Somerville NJ 08876. CIRCLE INQUIRY NO. 204

- **Find it Fast.** Micro Yellow Pages (formerly TRS Yellow Pages) is a 20-page newsletter/catalog on business software packages. Featured are an integrated accounting package for TRSDOS (Mod-II) and CP/M and packages for MBasic on Heath computers. Micro Architect Inc., 96 Dothan St., Arlington MA 02174. CIRCLE INQUIRY NO. 205

- **Fiberoptic Communications.** Fiber Topics, newsletter on new products, applications and technology advances, covers developments in the telephone, broadcasting, CATV, military and data communications markets. Valtec, 99 Hartwell St., West Boylston MA 05183. CIRCLE INQUIRY NO. 206

- **Test Instruments and Transmissions.** A 12-page catalog lists the entire Halcyon line of data communication test instruments and data transmission equipment. Also included is a complete listing of Halcyon sales and service centers. Al Miller, Halsyon, Inc., 2121 Zanker Rd., San Jose CA 95131. CIRCLE INQUIRY NO. 207

- **Information Systems.** A 12-page color brochure describes the Cado Systems Corp.'s equipment including a one-station system with video display/keyboard, a printer and floppy disk storage, and a multitasking configuration with 52 million characters of Winchester storage. Software is also described. Cado Systems, 2771 Toledo St., Torrance CA 90503. CIRCLE INQUIRY NO. 208

·MODEL 800 · WITH SUNFLOWERS·

A NEW MASTERPIECE IN PRINTERS

The MODEL 800 MST is certainly pleasing to look at, but its true beauty lies beneath the surface. A glimpse at its features reveals why it is rapidly becoming the most sought after printer in the world . . .

- Four standard interfaces:
  - RS-232 (15 baud rates)
  - Centronics compatible parallel
  - IEEE-488
  - 20ma current loop
- Six line densities: 64, 72, 80, 96, 120, 132
- 100 CPS at all six densities
- Unidirectional or bidirectional printing
- Sixteen horizontal and ten vertical tabs
- Elongated characters in all six densities
- 1920 character buffer
- Uses either perforated or roll paper
- Fully adjustable tractors to 9½"
- Auto self-test
- Up to 10 character fonts
  - Standard 96 character ASCII
  - User defined character font
  - Provision for up to eight additional fonts
- Dot resolution graphics in six densities
- Variable line spacing control from 0 to 64 dots in half-dot increments
- Auto form-feed for any form length at any line spacing
- Heavy-duty all aluminum chassis
- 110vac or 220vac, 50/60Hz.
- 100 million character printhead
- Measures only 15" wide, 3" high, and 11" deep
- Weighs only 15 lbs.

. . . . . but maybe its most attractive feature is the price . . . . . $699.00.

base 2. inc. P.O. BOX 3548 FULL., CAL. 92634 / (714) 992-4344

CIRCLE INQUIRY NO. 4


TERRI

Purchase Order

by W.B. Goldsmith, Jr.

Most of us buy merchandise by mail. We order books, software and kitchen utensils from across the country. Businessmen order many goods and services on some type of written communication. My family includes some dedicated mail-order junkies—in fact, I may be the most dedicated of all. I have submitted orders on 3x5 cards, the backs of envelopes and purchase order forms furnished by vendors. My personal records and files were in chaos as I tried to organize all these forms and lists of paper.

A program entitled Purchase Order solved some of my problem. My orders are now uniform and I can easily keep track of them. (My files are still a national disaster but that may pose too big a problem for any computer.)

The Basic routine provides a general purpose purchase form for business or personal mail ordering. It is of professional quality, provides all the information required by a supplier, and organizes the data uniformly for your files. If your printer doesn't produce carbons, you can make multiple copies at the touch of a keystroke. You can customize this program to fit your personal situation with very little strain, and use it as a starting point to produce an elaborate version if you need more purchase order power.

User's notes

As the sample run demonstrates, the program prompts all the necessary data inputs. All entries except "city, state zip" are a one prompt-one entry situation. The entries for "city, state zip" are one prompt—two entries to allow you to type the city (comma) state and zip following normal habit patterns.

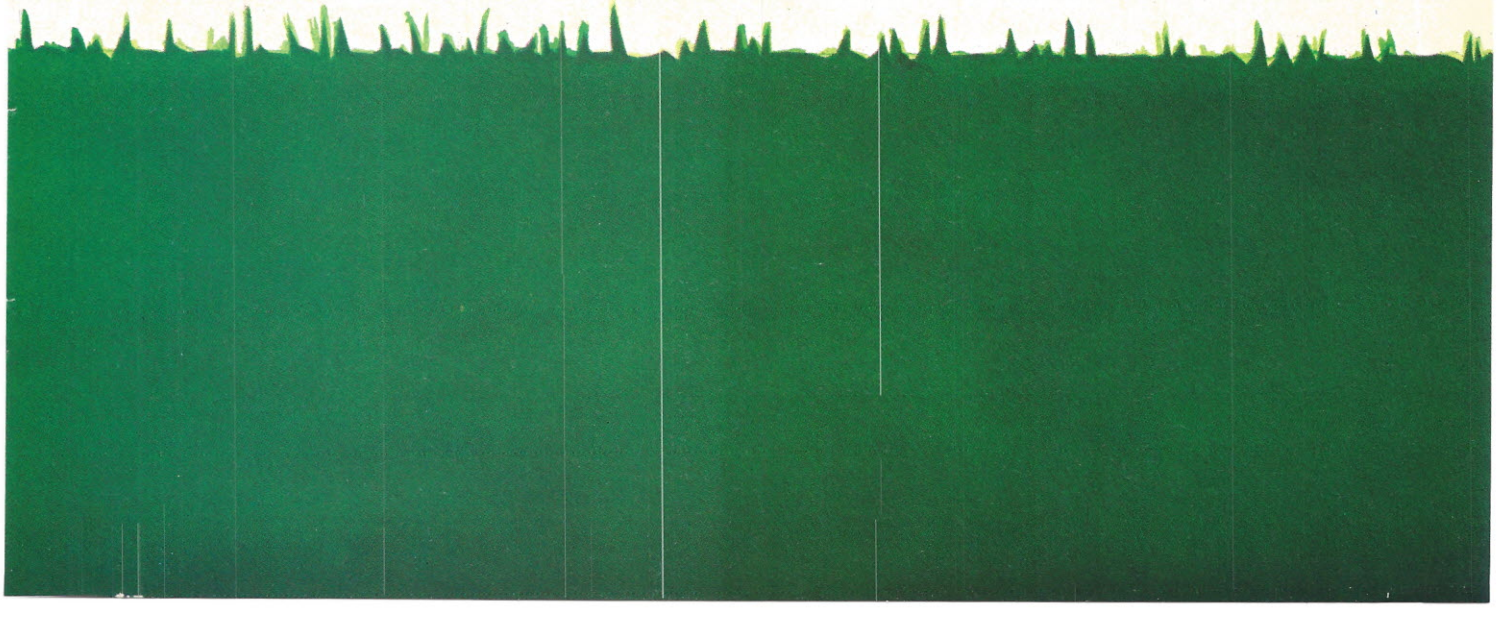
Before you start, decide how many different items you want to purchase from a particular vendor since that is one of the first questions the program asks. Most of us plan our orders before we reach for the old checkbook, so this bit of preplanning is not too restrictive. In any event, a practice run will make you a master user.

The program

Purchase Order is written in SWTP 8K Basic 2.0. It should adapt readily to other Basics having string variables and one dimensional arrays. Multiple statements are used with a colon separating each part. If your interpreter won't handle multi-statements, just insert separate line numbers. There should be sufficient freedom in the line numbering scheme to permit extra insertions.

Statements 100 through 120 are function definitions used to simplify print formatting. You won't need these if you have a print format function in Basic. My TTY has only 72 column print capability, so all the numbers are based on formatting for that width. Line 100 is a centering format, while 110 and 120 line up the decimal points for the money columns.

The SWTP interpreter restricts string variables to 32 characters each. In statement 130, the restriction is changed to 50 characters. (You can change the variable capacity from 7 to 72 characters with the 'poke.'



I've opted for a variable length of 50 with this application, because the first PO I processed needed it.) This poke line is unique to the SWTP Basic, so delete it for any other—or insert the appropriate version of variable length designator for your software. Similarly, statement 160 defeats an automatic CR LF feature in Basic and permits print format control by the programmer. Drop this one, too, if you don't need it.

If you always use the same name and address on your letterhead, you can remove lines 200 through 230 and enter your constant strings in 510 through 530 like this:

```
510 PRINT TAB(35);"ABC"
520 PRINT TAB(29);"123 ANY STREET"
530 PRINT TAB(28);"UTOPIA, CA 98765"
```

Additionally, for personal POs delete 960 which prints the business name below your authorizing signature.

The question in line 380 drives the calculation of a sales tax or printing of a tax resale number. For non-business applications, you may wish to delete 400, 410, 485, and 490; and change 380 to 'input "sales tax exempt",Z$.' If sales tax isn't a problem with you (because you live in a non-sales tax state or always order out-of-state), delete 380, 390, 400, 410, 485, 490, 865 and 870. If you want to retain the sales tax feature, check the rate used for the calculation in line 865. California has a 6% rate and that's what I've used. Put in the applicable percentage for your jurisdisction.

Lines 430, 450 and 970 control the I/O device on my SWTP. My video terminal is the control terminal at port 1, and I have a TTY at port 3 which I use for a system printer. With different versions of Basic, you may need to rearrange these ' "port = " ' commands.

Another modification you may wish to try is calculation of the shipping/handling allowance as a fraction of the total dollar amount of merchandise ordered. To do this, delete line 370 and add a substitute line at 835 like this:

```
835 S=n*T ;
```

where n is the factor you wish to use.

To remit payment by charge card instead of cash or check, change 890 from 'print"total (enclosed)";' to 'print"total";' and add a sequence similar to:

```
911 PRINT
912 PRINT"CHARGE TO 'bankcardname' "
913 PRINT"NUMBER 123 456 789 000"
914 PRINT"INTERBANK NUMBER 9999"
915 PRINT"EXP DATE 09 90"
```

You should use the name and numbers for your own credit card, obviously.

The rather complex tab function in statements 950 and 960 provide centering of the authorizing name and company name under the signature dotted line. For your personal PO, you may wish to use a variation of:

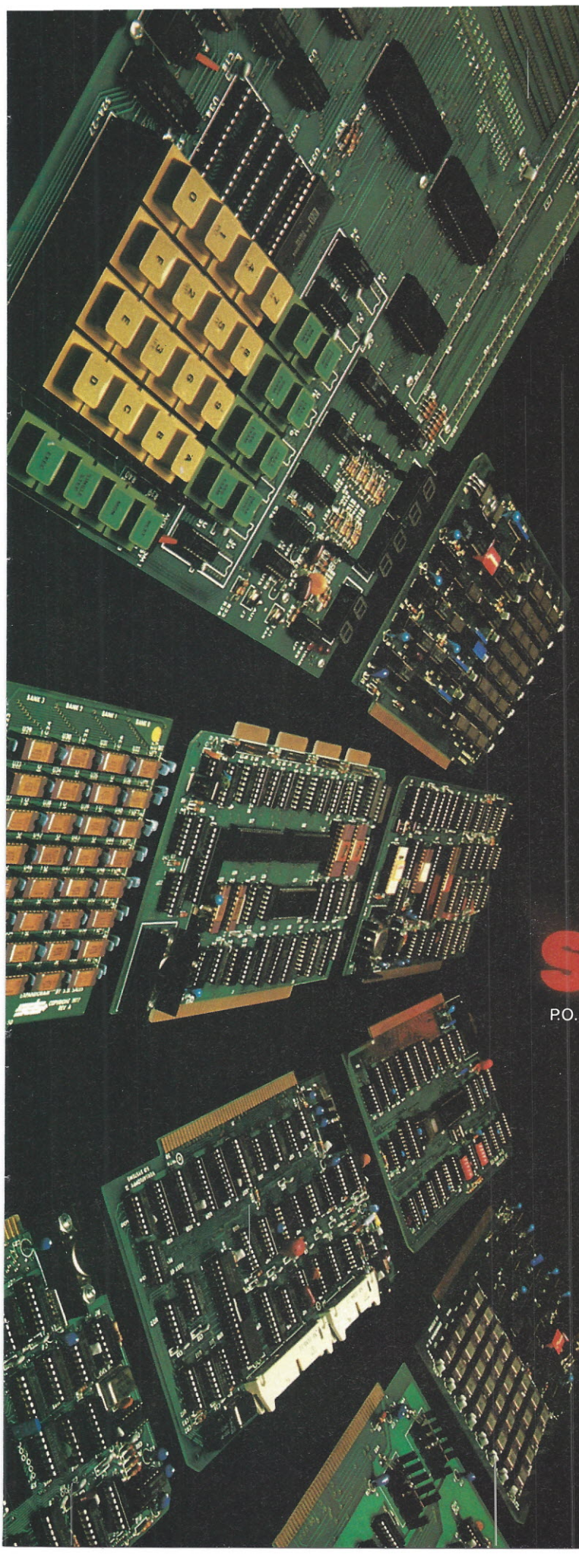
```
950 PRINT TAB (42);"I. M. BUYER"
960 PRINT TAB (40);"IMB ASSOCIATES"
```

Lines 980 and 990 allow another go through the print sequence. The vector to line 430 allows a change of output device for each subsequent printing. You can preview your PO on the video terminal before requesting a hard copy, or use one printer for the external copy and another for your file copy. □

Program on Page 124

Sample Printout

```
P.O. NR: 80-112                          RESALE TAX NR: 12345XTX

                    BUD'S COMPUTER WORKS
                      #1 SOFTWARE LANE
                      UTOPIA, CA 99999

                          20 MAY 1980

TO: COMPUTER SUPPLIES COMPANY
    123 MAIN STREET
    CORNUCOPIA, NY 11111

PLEASE SHIP THE FOLLOWING TO US:

QTY          DESCRIPTION              UNIT PRICE     EXTENSION

10       10 SECTOR MINIDISK              3.50          35.00

1        8 1/2 INCH PAPER ROLL           8.40           8.40

12       TTY RIBBON                      2.25          27.00

2        DISKETTE HOLDER                 5.00          10.00

1        DISKETTE ERASER                34.69          34.69

                                                    ----------
    TOTAL FOR MERCHANDISE                           $ 115.09

    ALLOWANCE FOR SHIPPING/HANDLING                   12.00
                                                    ----------
TOTAL (ENCLOSED)                                    $ 127.09
                                                    ==========

    THANK YOU!

                                  ----------------------
                                       I. M. OUNER
                                  BUD'S COMPUTER WORKS
```



every smart computer needs an SD SYSTEMS heart.

We design and manufacture a complete line of industry compatible microcomputer boards and kits that can serve as the heart of your system. All are S-100 Bus compatible and use the Z80 microprocessor.

MPC-4 —This SD Systems exclusive is a multi port controller which uses the Z80 for multi-user operations offering four serial RS-232 I/O channels.

SBC 100/200 —A 2.5/4 megahertz range of single board computers which are effective standing alone or combined with the complete SD board range.

ExpandoRAM I/II —For use with 250/200 nanosecond RAM, these high density boards offer 16 to 64K memory; the ExpandoRAM II can achieve RAM capacities up to 256K using 64K chips.

Versafloppy I/II —A floppy disk controller for up to four drives, supporting single/double density and single/double-sided disk formats.

VDB-8024 —A full function visual display board with a Z80 controller that adds display capabilities to your system.

Prom 100 —A specialty board of SD Systems which allows you to program 2708/2716/2732 proms.

Z-80 Starter Kit —A low-cost entry into the world of microcomputers designed primarily for education and experimentation.

SD SYSTEMS

P.O. Box 28810 • Dallas, Texas 75228 • 214-271-4667 • Telex 6829016

CIRCLE INQUIRY NO. 54

NOW YOU CAN SAVE $25 PER BOARD*
when you purchase any SD Systems microcomputer board from participating SD Systems dealers listed below. *Offer expires 10/31/80

ADVANCED COMPUTER PRODUCTS, INC.
Irvine CA • 714-558-8813

ANCRONA
Culver City CA • 213-641-4064

COMPUTER CITY
Charlestown MA • 800-343-6652 or 617-242-3350

THE COMPUTER MART
Waltham MA • 617-899-4540

COMPUTER PRODUCTS STORES
Springfield IL • 217-528-0027

CUSHMAN ASSOCIATES
Wilmington DE • 302-995-6733

DAL-COMP
Dallas TX • 214-350-6895

FUTURE ELECTRONICS
Natick MA • 617-237-6340

JADE COMPUTER PRODUCTS
Hawthorne CA • 800-421-5500

MINI MICRO MART
Syracuse NY • 315-422-4467

PRIORITY ONE
Sepulveda CA • 800-423-5633 or 213-894-8171

S-100
Clark NJ • 201-382-1318

Q.T. COMPUTER SYSTEMS, INC.
Lawndale CA • 800-421-5150 (ex. CA) or 213-970-0952

For complete product information, send for SD Systems' board and kit brochure (BK-101).

FAST SIMULATION OF NERVE POTENTIALS

by Dr. James E. Randall

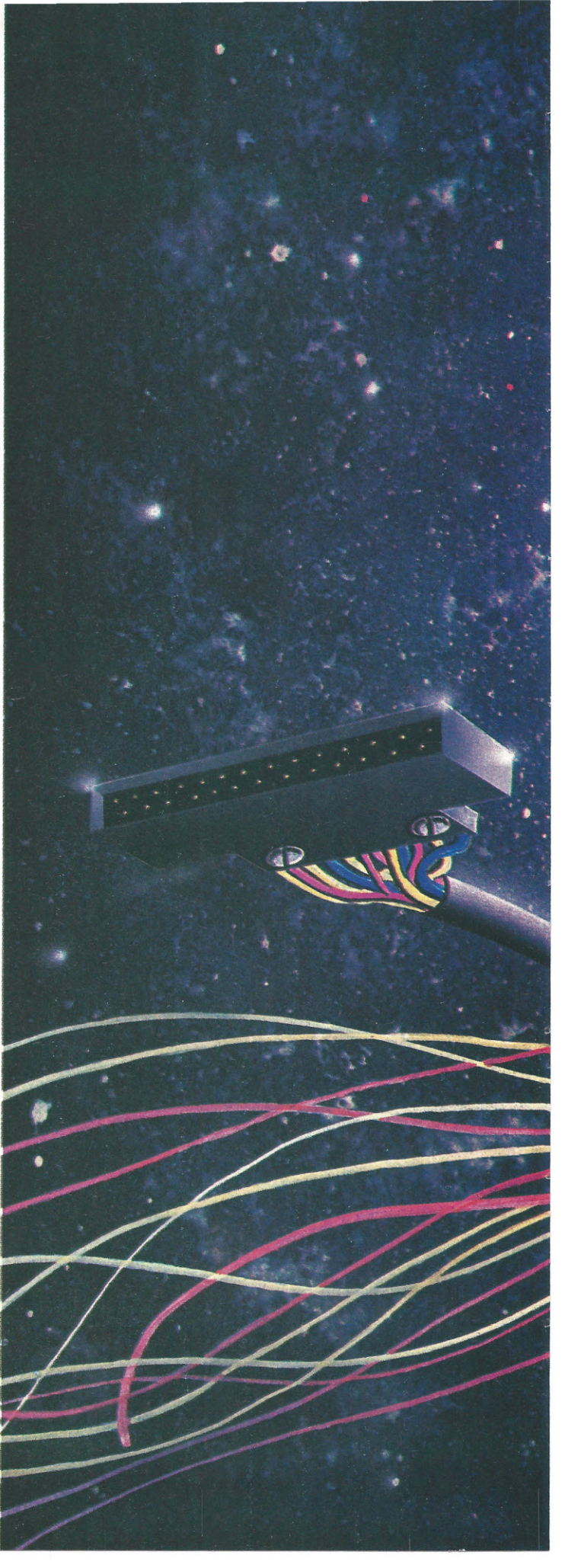
One of the attractive features of using the Apple II microcomputer for mathematical simulations is that it provides graphic displays with reasonable resolution at a modest price. However, when the algorithm involves the evaluation of several transcendental functions such as exponentials, the computation time using interpreter versions of Basic may become prohibitively long. The Am9511 arithmetic processor unit, a floating point numerical processor now available on Apple II interface cards, uses a benchmark program for speed comparisons. It is a neurophysiology teaching exercise. In this case, the program runs twice as fast if the Am9511 replaces the general Basic software subroutines and ten times as fast if user-written subroutines are written for the numerical parts of the simulation.

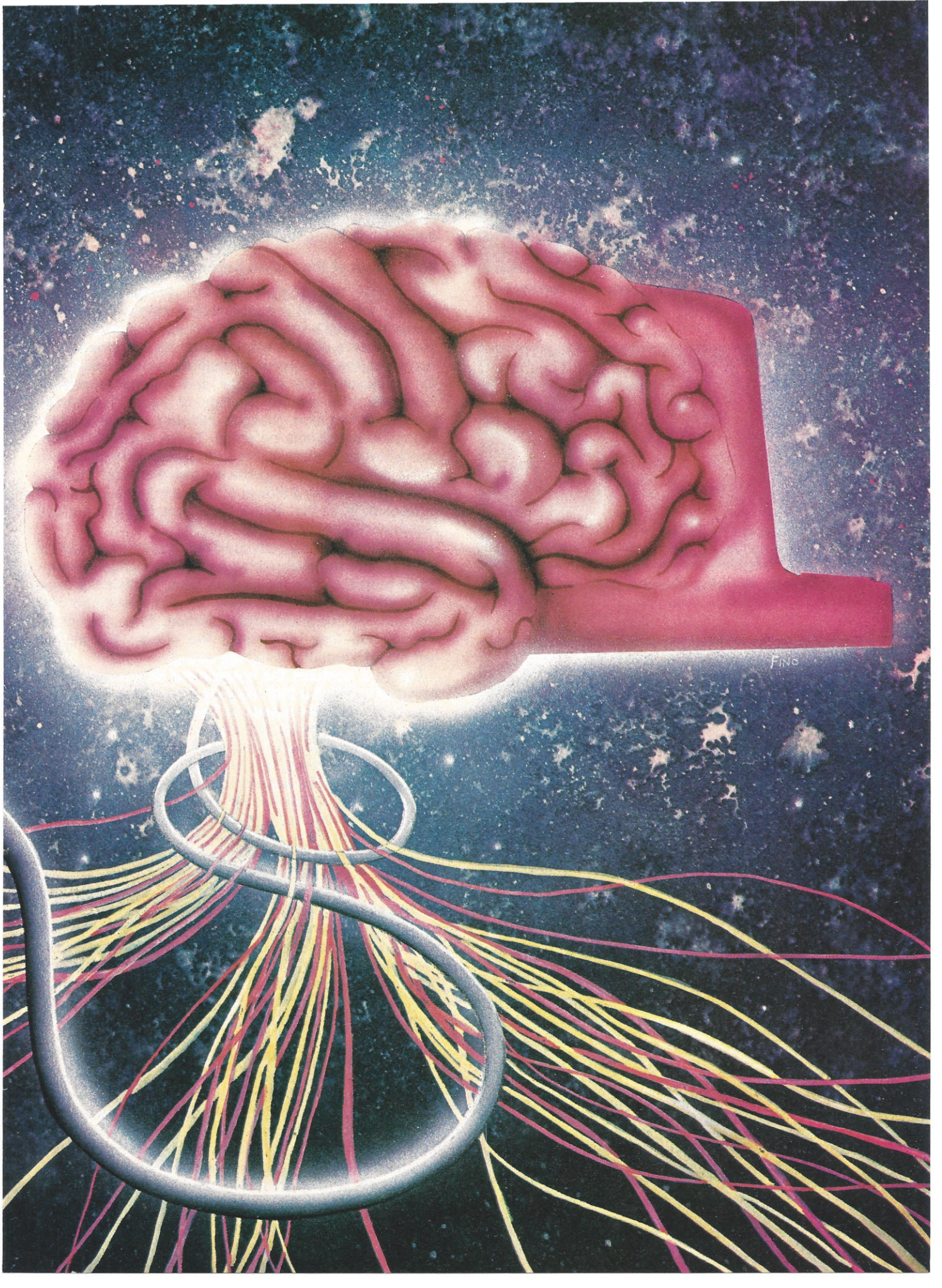
Nerve action potentials

The simulations described are used to teach first-year medical and graduate students the electrical mechanisms by which nerve cells communicate information between the peripheral organs and the brain. The axon extensions of these cells support a mechanism for sending electrical signals over long distances without attenuation. At rest, there is an imbalance of ions across a thin dielectric membrane that separates the axon interior from its environment. This results in a resting electrical potential difference of about −90 millivolts with the excess negative charge being inside the cell.

Information is propagated along the axon in the form of a transient disturbance of the membrane voltage, called an action potential or, because of its short duration, a spike. The magnitude of a sensed stimulus such as touch, heat, light, or sound is transmitted by both the number of activated nerve axons and the firing frequencies of their respective action potentials. Likewise, the force developed by muscle contraction depends upon how many of the nerve fibers that supply it are active at any instant.

In order to initiate the action potential, it is necessary for a stimulus to reduce the resting potential to a critical threshold value. The top tracing in figure 1 is a photograph of a nerve action potential as simulated on an Apple II for the conditions of a sudden change in membrane potential from −90 to −70 mv. This change of 20 mv triggers the much larger regenerative change from −70 to +20 mv. A response greater than the stimulus is equivalent to a voltage gain, the attribute that permits the signal to spread along the nerve axon without being dissipated.




FINO

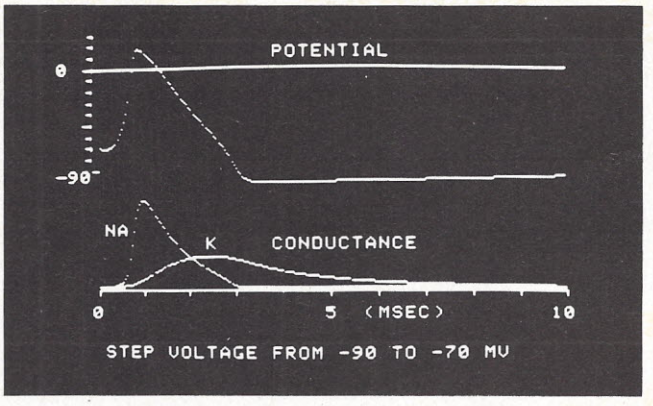


Figure 1. Response to step change in membrane voltage.

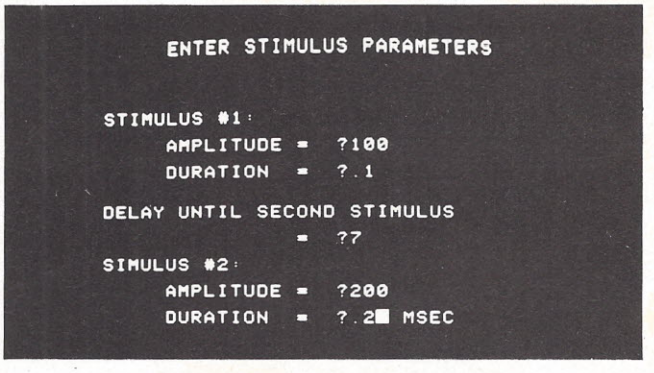


Figure 2. Entering stimulus parameters.

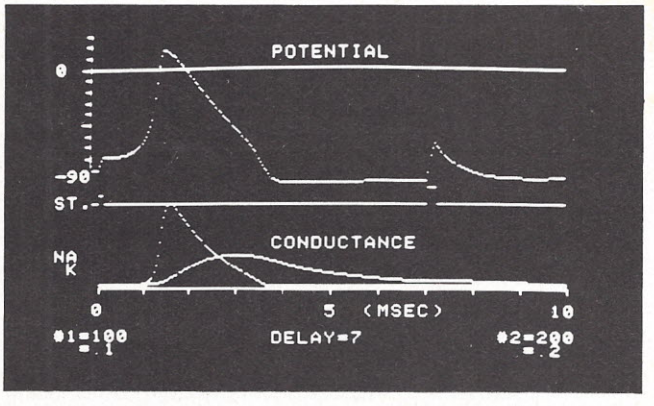


Figure 3. Response to parameters of figure 2.

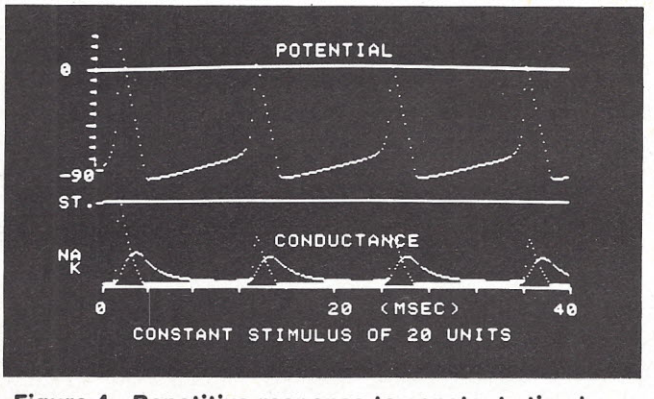


Figure 4. Repetitive response to constant stimulus.

One of the classic neurophysiology experiments is the demonstration of action potentials recorded from long nerve axons which are excised from anesthetized frogs. Electrical stimuli of known amplitudes and durations are applied to one end of the nerve and the evoked action potentials are measured at the far end. In this way, physiologists have measured the propagation velocity of nerves and cataloged the factors that determine the threshold for firing and the firing frequencies. Computer simulation provides students with an opportunity to observe these properties and also to delve into their mechanisms.

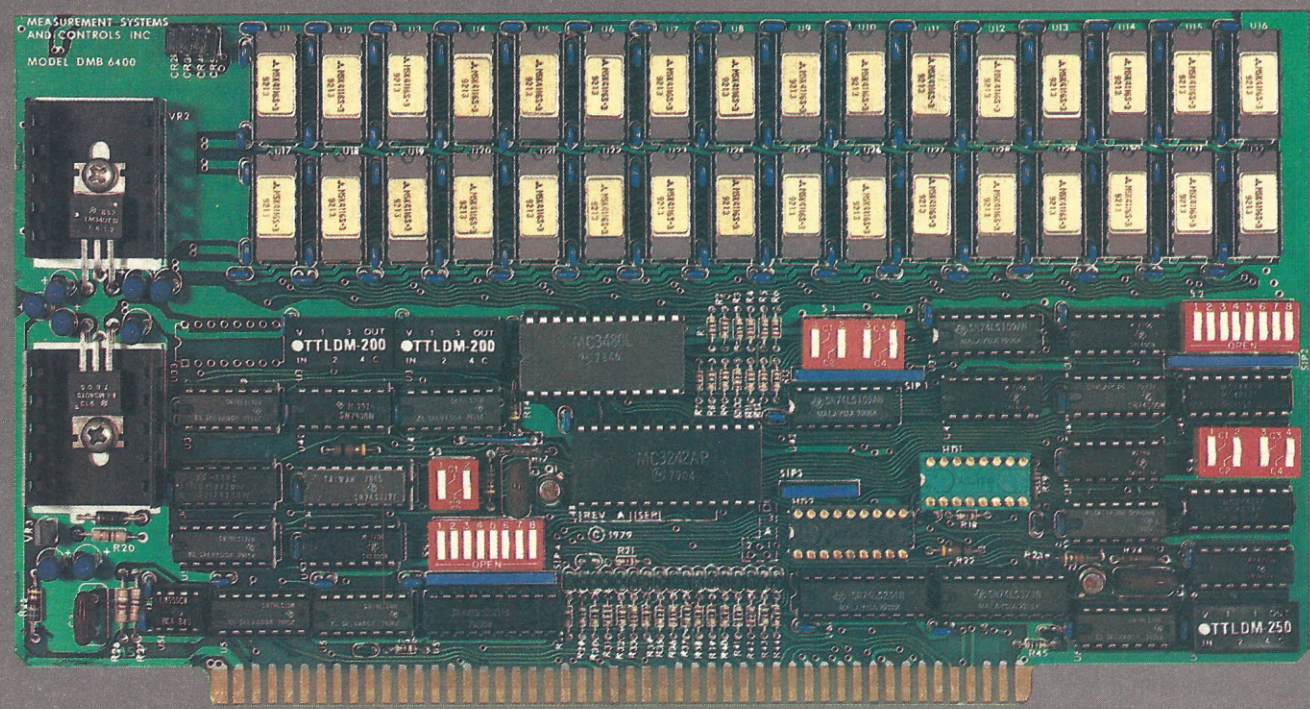
Simulation of the action potential

In 1952, two Englishmen, A.L. Hodgkin and A.F. Huxley, formulated a mathematical model that reproduces the axon's electrophysiological properties. The central theme of their work is that the membrane conductance (the reciprocal of resistance) for sodium and for potassium ions are time- and voltage-dependent functions. The bottom tracings of figure 1 illustrate these conductance changes initiated by the sudden change in membrane potential. There is a rapid rise in conductance to sodium (Na +) that allows these ions to enter the cell, reducing the negative charge therein. This is followed by a slower increase in conductance to potassium (K +) that allows these ions to leave the cell and restore a net negativity.

The stimulus for a simulated nerve response may be entered into the program as a sudden change in membrane potential from the resting value. Such is the case in figure 1. Another method is to have a selected current applied for a selected duration. The product of current and time determines the charge transferred and thus the amount of voltage change.

Figure 2 contains the parameters as entered by an operator in which a first stimulus was 100 units for 0.1 msec followed after a delay of 7 msec by a second stimulus of 200 units for 0.2 msec. The upper tracing in figure 3 is the computed response of membrane potential that results from the chosen stimuli plotted in the middle tracing. The waveforms at the bottom show the change in ion conductances predicted by the model. During each stimulus, there is a ramp change in membrane potential. For the first one, the initial voltage change of 10 mv is able to trigger the membrane conductance to sodium ions that cause further voltage change. The second stimulus, being twice as large and twice as long, changes the resting membrane voltage by 40 mv. However, at 7 msec the membrane is still unresponsive because of the aftereffects of the first response. The same stimulus combination separated by 8 msec will elicit two responses. Students may map out the threshold required for different delay times and thus learn by experimentation of the refractory period of nerve.

A second example of refractoriness is indicated by the time between successive spikes which is a function of the amplitude of the stimulus, i.e., its analog/digital conversion properties. Figure 4 is a 40-msec sweep in which the stimulus amplitude is set at a value of 20 units for the whole 40 msec. Initially, the stimulated membrane is very responsive and the action potential is initiated at a membrane potential of about -85 mvolts. After the spike, when the nerve is refractory, it takes the stimulus about 10 msec to get up to an ele-



BANK SELECT — 64K BYTE EXPANDABLE MEMORY BOARD

MODEL DMB6400 SERIES

FULLY COMPATIBLE WITH:

ALPHA MICRO

CROMEMCO

NORTH STAR

MP/M

and most other S-100 systems

- Four independent, 16K software selectable banks.
- Switch selectable bank sizes — from 16K to 64K in 16K increments.
- Eight banks (512K) per I/O port for each of the 256 ports.
- Z-80 4MHz operation with no wait states.
- Low power — 8 watts maximum.
- Reliable, tested and burned-in memory.
- **ONE YEAR GUARANTEE**
- IEEE S-100 compatible timing.
- Attractive Dealer & OEM Prices

MEASUREMENT systems & controls incorporated

867 North Main Street • Orange, CA 92668
Telephone: 714/633-4460

vated threshold of −75 mv. This is a repetitive process with 4 spikes occurring within the simulated 40 msec, corresponding to a firing frequency of 100 spikes/sec. Experimentally, students may calibrate the firing frequency as a function of stimulus amplitude.

Simulation in Applesoft Basic

The appended program listing has the essential steps of an Applesoft Basic program that will plot membrane potential and conductance changes following selected stimuli. For brevity, many steps have been omitted, such as placing axes with 'hires' characters on the screen or formatting the input statements.

Lines 500-570 initialize parameters while the main iteration loop is in lines 1000-1050. There are four subroutine modules. The routine at line 2000 sets six first-order rate constants, denoted by the symbols alpha and beta by Hodgkin and Huxley, according to empirical functions of the membrane potential. Each of these evaluations involves an exponential, the time-consuming part of the simulations. The routine at line 2500 uses the rate constants to determine the initial values of Hodgkin and Huxley's dimensionless factors, N, M, and H, numbers lying between 0 and 1 that indicate the efficiencies of the sodium and potassium conductances for each iteration. Lines 4000-4100 update the membrane currents to find the change in membrane potential during an integration interval. The stimulus is added in if the time variable (T) falls within the stimulus duration (SD). The subroutine at line 5000 plots the variables upon the Apple II 'hires' screen.

Replacing Apple math subroutines

The 120 seconds required to run the program is not excessive for one or two runs. However, if a student is expected to try a whole family of stimulus parameters or to simulate 40-msec sweeps, which take eight minutes each, computation speed does influence the effectiveness of the teaching exercise. The easiest way to use the Am9511 numerical processor to increase speed is to replace the basic floating point math subroutines with calls to the hardware floating point processor.

In 1979, two manufacturers marketed Apple II interface boards which use the Am9511. California Computer Systems of Santa Clara, CA makes the model 7811B Apple arithmetic processor, and Computer Station of Granite City, IL sells a fast floating point board, model 7001. Both come with diskettes containing RAM-based Applesoft in which the Am9511 is used for evaluating the math functions. The usual nine-digit precision of Applesoft is reduced to about seven digits but the speed of the Basic routine is increased by a factor of two to three times, depending upon the computations involved. The advantage of this approach is that the user retains the conveniences of the Basic interpreter.

Further speed increases require removal of the rate-limiting Basic interpreter. As compiled languages, such as Pascal and Fortran, become popular with the Apple, it will be possible to have versions based upon a hardware floating point processor. The alternative chosen by the author for the nerve action potential simulations was to write a set of assembly-language subroutines which can be called from Basic. The view was that the Hodgkin-Huxley model for nerve and the Apple II are popular enough to justify the one-time effort of developing the machine-coded simulations.

Simple classroom applications

The 6502 microprocessor nerve simulation subroutines developed are general enough that neurophysiology instructors can write their own Basic routines, stressing teaching objectives by text displays without having to reassemble the machine code. For example, the simulated time range is set in integer multiples of 10 msec by a 'poke' command that indicates the number of computation loops per plotted point. Similarly, the integration time increment can be set from Basic. Once the parameters of the simulation have been obtained by 'input' statements, the machine-language routines use them to execute the simulation using the Am9511 to its full advantage. The high-resolution graphic routines inherent in Applesoft are used for the plotting. After 250 points are computed and displayed on the CRT, the program returns to Basic for the next exercise. This approach reduces the computation and plotting time to 10 seconds for the 10-millisecond sweep. Axes and 'hires' labels, applied with Basic, require an additional two seconds. Any further increase in the speed of computation using current microcomputer technology would require faster clock frequencies for the 6502 microprocessor and for the Am9511 numerical processor.

As compiled languages. . . become popular with the Apple, it will be possible to have versions based upon a hardware floating point processor.

If a programmer wishes to utilize a combination of Basic and machine-coded subroutines which utilize the Am9511 numerical processor, it is wise to first write an all-Basic version, and then introduce and debug the 6502 routines one at a time. The author first wrote a routine that moved the Apple floating point accumulator from its page-zero location to a buffer area where its format could be examined by 'peek' statements. Then subroutines were written that made the minor modification of the Applesoft floating point format to that of the Am9511. Arguments were transferred from Basic to the Am9511 and back to Basic. 'Poke A,B' commands were used to issue opcode B to use the Am9511 command port that was memory-mapped at address A. Such operations revealed the truncation errors in which a value such as 0.01 was returned as 0.00999999. Once the numerical processor subroutines were functioning, the simulation modules were developed and tested individually using the Basic-Am9511 software interfacing routines. Finally, the whole binary package was saved by a 'Bsave' command to provide the file that was loaded by the first Basic statement. □

Program on Page 125

YOUR INVITATION TO COMPUTING ADVENTURE

INTERFACE AGE HAS EVERYTHING YOU WANT TO KNOW ABOUT COMPUTING

Every issue is written and edited by experts. Every issue brings you how-to's and know-how on:

- Software and Hardware
- Tutorials on all phases of computing technology
- New Product Design and Reviews
- Games, Robotics and Energy
- Educational and Developmental Applications
- Business, Accounting, Investment Analysis, Law and Medical Applications
- Hardware and Software Reviews
- Personal and Home Projects

It's a 12-month program to help you enjoy all computing activities no matter what your special interest is. All the spectacular computing fun, adventure and know-how every computerist dreams about and it's all yours. We guarantee* it.

DON'T WAIT! SUBSCRIBE TODAY!

***Unless you are completely satisfied with INTERFACE AGE, you may cancel your subscription at any time and receive a full refund for all unmailed copies.**

EXCLUSIVE 1980 BONUS

INTERFACE AGE brings you a complete course on "How to Build and Use a Micro Computer Based System." This series will cover all aspects from system development principles to how to use a database management system. In addition, several interesting installments will be devoted to providing the reader with a good grounding in the FORTRAN language.

DON'T MISS A SINGLE ISSUE!

Please enter my subscription for: **9/80**

☐ 1 Year (12 Issues) $18 ☐ 1 Year Canada/Mexico $20
☐ 1 Year Foreign Surface $28 ☐ 1 Year Foreign Air Mail $50

☐ **Enclosed is my check or money order in U.S. Funds drawn on U.S. bank.**
Make check or money order payable to:
INTERFACE AGE Magazine, P.O. Box 1234, Cerritos, CA 90701
In Europe send to:
INTERFACE AGE EUROPE
Dahlienstr. 4, D-8011 Munchen-Vaterstetten, West Germany

☐ Bill Me Please charge my ☐ AMERICAN EXPRESS Cards Welcome ☐ VISA ☐ master charge THE INTERBANK CARD

Signature _____ Expires

RUSH my copies to:

First Initial		Middle Initial		Surname	
Mail To:					
City			State	Zip Code	

Beating the System

by Mark J. Borgerson

Every programmer reaches the point where his computer just won't do what needs to be done. That's when to start trying out ways to beat the system. In my case, the system is part of the Apple Pascal package. The problem was: how to run one of three large programs at the request of the operator.

The programs are too large to store in the computer memory at the same time. Since they are part of a package to be used by a businessman to maintain records of client transactions, it shouldn't be necessary to learn the complete Pascal language. He should merely select an appropriate program from a menu; then load and execute it.

Unfortunately, the Pascal system only wants to load and execute programs like the Compiler, Editor or Linker. There is a command (execute) that allows any program to be loaded and run, but it requires the program name to be spelled out. It doesn't provide a menu listing all possibilities.

A possible solution

If I had written the program in Basic, the solution would be simple: write a short program to list the menu, input a number representing the program desired, and use the 'chain' command to load and execute it. The Pascal system, however, does not have a direct analog to the Basic chain command. The closest is the 'segment procedure' facility.

I decided that the operating systems as originally set up would not do what I wanted.

Segment procedures are routines loaded into memory only when required for program execution. However, like all procedures in Apple Pascal they are limited to about 1200 bytes of object code. Thus, each part of the program would have to load several segment procedures into memory to complete its assigned duties.

Each procedure used by all program segments would have to be set up as a forward procedure. In addition, for segment procedures to operate, the disk with the program must be in the drive whenever the procedure is called. Since many procedures are not needed until a particular operation is requested, they would not be loaded until then.

Because the program normally uses both disk drives for data files, I included halts and prompts for the diskette throughout the program when one of the segment procedures was first called. This is necessary at each subsequent call of the procedure, since it might be overwritten by other segment procedures called when the first was not active and only seven are allowed.

After a study of the reference manual, phone calls to a number of experienced Pascal programmers and to Apple Computer's hotline, I decided that the operating system as originally set up would not do what I wanted. So I set out to beat the system.

Course of action

My thinking went something like this: In the normal system, utility programs are selected from a menu with a single character. The system accepts a single character, then loads and executes the desired program. I decided that I needed to fool the system into loading and executing my programs instead of the Editor, Compiler or Filer.

The easiest approach was to simply put my data entry program on the disk, delete the old System.Editor program and rename my program System.Editor. The system would still prompt for the user to select from among editors, compilers and filers: but the programs loaded and run would be those I had written. However, this would be confusing for users trying to distinguish between the old prompt line and the new programs.

What I really wanted was to get into the System.Pascal file and change the prompt string to match the programs. Since I was modifying the file, I changed the file names, primarily to keep track of files on the disk and minimize confusion (confusion costs my customers money).

All right, I'm going to make some small changes in the System.Pascal file. How do I do it? If I worked at Apple Computer, I would simply put the diskette with the source code for the System.Pascal program into my disk drive, run the Editor, change the source code, then recompile the program. The new object file would be transferred to the customer's system disk, the disk transferred to the customer, and the customer's check transferred to my pocket.

No inside help

Alas, I work in a computer store in Oregon, not at Apple. For reasons known only to them, Apple has not released the source code for the System.Pascal program, which is only useful to someone who already has Pascal. This is especially true for Apple II because a 16K memory card is part of the package. Since I've now put in my two bits on the freedom of digital information, I'll get on with the solution.

The System.Pascal file is a P-code (pseudo-code); that is, written in Pascal with some earlier version of the current Pascal system or on a different computer. The character strings used by the program are contained as Ascii characters embedded in the file. These strings include the prompt line, various error messages and the names of all the files directly used by the operating system. The code file is approximately 17K bytes long.

Since I want to change the embedded strings, perhaps I should just use the Editor program. A simple solution—but it didn't work. Many bytes of P-code in the program cause the editor to do strange things on the screen. It is difficult to edit a file with a large number of seemingly random cursor moves, screen erases and backspaces.

What I really needed at this point was an editor designed to work with hexadecimal codes in files not necessarily text files. (In many minicomputer operating systems, programs like this are vital to make patches to the systems programs 3.12XX'' instructions. These patches generally don't improve system capabilities—they just correct the bugs found by the first 200 unsuspecting users.) Since a file patching utility is not offered, I had to write it myself.

Flexible program constructed

The source code for the resulting Hexedit program is presented in the listing. It allows you to select a file by name, view the contents in both hexadecimal and Ascii, then change any desired data words. I use the noun "words" on purpose: although data is dumped in 8-bit bytes on the screen, the program requires you to modify a complete 16-bit word when making changes.

The program was written this way for two reasons: the 16-bit word is the natural unit of data for Pascal; and it was easier this way. The program will present a menu allowing a choice between dumping the data in the file, modifying the data or terminating the program. In the display or dump mode, Hexedit displays the 16-bit address of data relative to the start of the file, then 8 bytes of data in hexadecimal format followed by eight Ascii characters.

The Ascii represents the printable characters in the file, while nonprinting characters are shown as periods. You are prompted for the beginning address of the dump and can terminate the dump part of the program at any time by hitting the ESC key. A control-P will route the dumped information to the printer and a control-Q will return the output to the CRT (figure 1).

The 'modify' portion of the program also prompts you for a 16-bit address, then displays the 16-bit word found at that address. If the address of an odd-numbered byte is specified, the address is rounded off and the requested byte will be in the second half of the data word.

Modifications byte-by-byte

If you wish to modify only one byte of a data word, you must reenter the old data for the other half of the word. After the carriage return completing the data word, the next data word is displayed. If a complete word is to remain unchanged, a carriage return with no other input will leave the word unchanged and move to the next word.

```
4468: 05 01 9E 04 00 B8 01 0A   ........
4470: 00 B8 01 0B 00 B8 01 0C   ........
4478: 00 AB 05 B2 01 46 D7 A6   .....F..
4480: 4D 43 6F 6D 6D 61 6E 64   MCOMMAND
4488: 3A 20 45 28 64 69 74 2C   : E(DIT,
4490: 20 52 28 75 6E 2C 20 46    R(UN, F
4498: 28 69 6C 65 2C 20 43 28   (ILE, C(
44A0: 6F 6D 70 2C 20 4C 28 69   OMP, L(I
44A8: 6E 6B 2C 20 58 28 65 63   NK, X(EC
44B0: 75 74 65 2C 20 41 28 73   UTE, A(S
44B8: 73 65 6D 2C 20 44 28 65   SEM, D(E
44C0: 62 75 67 2C 3F 20 5B 49   BUG,? [I
44C8: 49 2E 31 5D 20 08 AA 50   I.1] . .P
44D0: CD 00 27 EC 00 00 CD 00   . .'.....
44D8: 29 AB 04 CD 00 25 EB 3F   ).....%.?
44E0: C3 A1 40 B2 01 46 D7 A6   . .@. .F. .
44E8: 2A 43 6F 6D 6D 61 6E 64   *COMMAND
```

Figure 1. An example of the output from the Dump routine.

```
STARTING ADDRESS: 100↓

0100: 0000  1234↓

0102: 0000  EADC↓

0104:↓

0106: 3333  FFFF↓

0108: (ESC)

OPTION:   (Waiting for D, E, or M)
```

Figure 2. An example of the Modify routine. Carriage returns are shown as ↓ . Operator input is underlined.

An ESC character terminates the modify procedure and returns to the main menu. Error checking on input of data words allows only hexadecimal characters to be entered as part of the input. The backspace (back arrow on the Apple II) can be used to edit the input prior to the carriage return. An example of the modify routine in action is shown in figure 2.

Hexedit was just the tool I needed; now what to do with it? First I found the prompt string at address 31A3 in the System.Pascal file. I modified the string to display a menu fitting the programs my customer would be using. Next I found a table of file names at address 4480. The names for the compiler, editor and filer were modified to suit the new programs I would be adding to the diskette. I only changed the prompt string and the file names; I did not change the characters used by the system to select among these programs. Without the source code, it would take a lot of work to find and alter the 'case' statement used to make the selection.

Customer error easily reduced

Instead I wrote my display menu to use the original characters. While the prompt line shows only three possible choices, the responses for other programs are still accepted. I can use the new system to execute other programs and I simply instructed my customer to type a carriage return if an input error on his part results in an unexpected response. This will bring him right back to the main menu.

What have I gained from all this work? Primarily a satisfied customer. Also, I learned a little more about the Apple Pascal system and got a chance to share that knowledge. My customer gained the ability to bootstrap his Pascal system and make a clear and simple choice between three programs that do the tasks for which he purchased the computer. His computer now executes his programs with single-key commands instead of requiring the longer and more complex commands that would have been needed if I didn't. . . beat the system. □

Program on Page 126

WHEN OPPORTUNITY KNOCKS ...

Open the door, and experience the dream. Your own business, the one you've always yearned for. No more limits on your creativity or earning potential. No more working for someone else.

If you're a successful computer professional and have access to some capital, chances are you could own a MicroAge Computer Store. MicroAge Computer Stores are built around a whole new concept of what a computer store should be. Designed to provide solutions, not just sell hardware.

MicroAge Computer Stores are firmly positioned in the business and professional marketplace. And as a MicroAge Computer Store owner, you'll be backed by one of the acknowledged leaders, industry pioneers in microcomputing. Contact the Director of Franchising today to receive a complete Franchise Information Package free of charge and at no obligation. Opportunity is knocking . . . you'll see what we mean.

MicroAge
COMPUTER STORE

1425 W. 12th Place • Tempe, AZ 85281 • (602) 967-1421

CIRCLE INQUIRY NO. 43

How to Solve Your Damaged Disk Dilemma

by Gene Cotton

What do you do with those 8-inch floppy disks with unreadable spots in the most awkward places? If you are reluctant to throw them away (and occasionally forget and use them again), here is a possible solution.

The mainframers mark bad tracks and substitute alternate tracks. CP/M does not, and setting aside tracks when you only have 77 to start with does not sound attractive. The alternative would appear to be to gather up the bad spots and put them in one file which would then keep the bad spots out of the available disk storage space. This approach would be consistent with CP/M as long as the file directory entry is a CP/M file name, and the disk storage allocation is done in compliance with CP/M operating convention. This leads to writing a program which will read all the records on a diskette and place those records which are unreadable into a special file called '[unused].bad' consistent with the CP/M conventions.

Saying is always easier than doing. The reason we have trouble getting the program off the ground is because we don't understand what is implied in words like "records," "file" and "CP/M operating convention." After these concepts are better understood, the problem of writing a program can be tackled. Therefore, two statements are proposed:

Investigate the inner workings of CP/M: Discover how CP/M keeps track of files, how disk space is allocated, what constitutes a CP/M record, and how records can be interrogated and classified as readable and unreadable. Also, when gathering up bad spots on disk and putting them in a single workable file, will the diskette be returned to use?

Write a program to gather bad spots into a single file: Assuming the answer to the final question of the statement above is yes, write a program that will group bad data records under the CP/M file '[unused].bad.'

This looks more promising. If nothing else, it should tell us if we are wasting our time.

The inner workings of CP/M

Listed below are some questions/tasks that need to be examined before any of the problems can be solved:

- How does CP/M keep track of files?
- How does CP/M allocate disk space?
- What is the basic CP/M disk record?
- What facilities are available in CP/M to read records, build file entries and note when a record is read in error?

CP/M segregates the disk into two distinct regions and three logical functions. The two regions are:

System Software—disk boot, CP/M, disk I/O routines
User Files—named files available to user programs

The user file area is further divided into two logical functions: File Directory Area—name and allocation information; Data Area—the actual user program data.

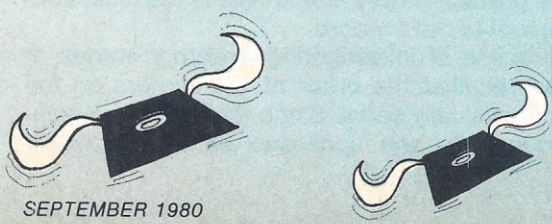
The physical organization of the diskette is detailed as 77 tracks of 26 sectors per track and 128 characters of information per sector. A track is a concentric circle on the face of the diskette. Tracks have a close analogy to the groove on a phonograph record. But, rather than spiral-shaped, the tracks are individual rings inside of rings inside of rings. Each track or circle is divided into 26 small arcs. Each arc is called a sector and is composed of a continuous stream of 128 characters.

The addressing scheme used to identify various sectors of data is to refer to the track on which the sector resides and then the relative number of the sector from an arbitrary beginning point on each track. (The beginning point is called the index and is identified by a physical hole in the diskette.) The tracks number from 0 to 76, with tracks 0 and 1 reserved for the operating system software (CP/M).

Tracks 2 through 76 are used for data files. The beginning of this data area is set aside for directory entries to identify the name and location of particular files in the remainder of the data area. Track 2 is used as the directory track and is setup to handle 64 directory entries. A directory entry contains the file name assigned by the user and the disk storage allocation map assigned by CP/M Basic disk operating system (BDOS). Each entry is 32 characters long and is composed of the following fields:

position	data type	content
0	Hex	Entry type (00 = used; E5 = unused)
1-8	Ascii	File name (space filled to the right)
9-11	Ascii	File type/extension (space filled to the right)
12	Hex	Extent number (directory entry number) (00 for 1st 16K, 01 for 2nd 16K, etc.)
13-14		Not used, but assumed hex zero
15	Hex	Record count (size of this extent) (0 to 128 sectors)
16-31	Hex	Disk allocation map (CP/M controlled)

A file may have as many as 16 directory entries (extents) to allocate space on the diskette. Each directory entry controls 16K of disk storage space; therefore, the 16 extents could allocate 256K (2048 sectors) of disk storage. It should be noted that this is sufficient, since a diskette has only 243.75K ((77-2)*26 or 1950



sectors) of data storage area (and some of that is taken up with directory entries). Sixty-four directory entries of 32 characters each means that 16 sectors or 2K of disk storage is set aside for the directory area. Approximately 241K of disk storage area is available to actually hold the data.

By looking through disk dumps of the directory area, it is discovered that the extent number field (position 12) specifies what extent (portion of the disk) this entry is controlling, and the record count (position 15) specifies how many sectors are controlled by this extent. If we just knew where the sectors were.

The disk allocation map

With another look through the disk dumps of the directory area, this time noticing the various file sizes and placements, the disk allocation map (positions 16-31 of the directory entry) begins to show a pattern. The disk data area is 243K characters long. A single byte, when viewed as an 8-digit binary number, may express values from 0 to 255 decimal.

By partitioning the data storage area into 243 1K blocks, each 1K block could have a relative placement number with respect to the beginning of the data area. In addition, that relative address could be represented by a single byte or character. With 16 bytes in each allocation map, 16K of disk storage could be identified. This means that the smallest block of disk data storage that can be allocated is 1K. If a file requires less than 1K (or several K plus some portion at the end), CP/M must still allocate the full 1K (or multiples of 1K) to the file. The drawback imposed by the 1K block allocation is compensated for in the ease with which the addressing scheme is implemented. As an added bonus, reclaimed disk storage space (as when a file is deleted) may be easily reallocated.

This scheme could produce a logical file which is not serially contiguous on the physical file. This is not a problem though, since CP/M BDOS is buffering between us and the data file. We see the file as a stream of logically sequential sectors and serially contiguous.

The sector as a record

It is obvious that these physical sectors are the smallest group of data which CP/M chooses to transfer between memory and the diskette. This brings us to the problem of defining a record. For the purposes of this investigation, it would seem justified to consider a record as a sector. Generally when we speak of "record," we mean the grouping of data "fields" which are associated with this information unit. An example is the name, address, city, state and zip code fields grouped together and associated with a larger information unit called a customer record.

Rather than force the concept of "record" into our problem, let us make an observation. The actual content of any data grouping is unimportant as long as the data is, in fact, readable and does not get lost in transit from the surface of the diskette to the internal memory of our computer. In short, the smallest amount of data that can be physically read at one time is the most convenient amount to call a record.

In the process of investigating directory entries, it was discovered that the order of the sectors on the track is not physically serial. Since the tracks are numbered sequentially and accessed serially, it would

seem natural for the sectors to be organized the same way, i.e., 1-26 serially. However, a problem can be seen with this arrangement. Suppose a sector is read and a modest amount of processing is done on that record, and now it is time to read the next sector. Alas, while the last sector was being processed internally, the next sector has rotated past the read head. Now we must wait for the diskette to make a complete revolution before the sector is again in position to be read.

A method of coping with this delay is to place the second sector farther around the track, so that it is more likely to be read without the extra disk rotation. This technique is called interlacing. For example, CP/M has placed the logical sectors 1, 2, 3,. . ., 26 in the physical sector locations 1, 7, 13, . . ., 22. The complete mapping is:

logical	physical	logical	physical
1	1	14	2
2	7	15	8
3	13	16	14
4	19	17	20
5	25	18	26
6	5	19	6
7	11	20	12
8	17	21	18
9	23	22	24
10	3	23	4
11	9	24	10
12	15	25	16
13	21	26	22

CP/M facilities for disk input/output

At first glance, the only facilities for disk transfer are through the CP/M BDOS disk access primatives. It is convenient to let BDOS open and close files. The actual disk location of the directory entry is not of interest to us. The allocated data area is our primary concern. If we are to locate the unreadable sectors and isolate them, we must control which physical sector is to be read. Without this capability, we will have little chance to find the bad spots. It is particularly troublesome to have BDOS trapping errors and shielding us from them.

CP/M BDOS is designed as a logical disk controller which depends on the Basic input/output system (BIOS) to perform the actual disk functions. This approach allows CP/M to be implemented on various disk drive configurations by simply changing the BIOS. Among services that BIOS provides are: selection of disk drive, physical track selection, physical sector selection and physical reading of sector.

Considering the design foresight which is evidenced elsewhere in CP/M, it is reasonable to assume that the interface between BDOS and BIOS is implemented in a standard way. A look at the CP/M system alteration guide confirms that the routines needed to selectively read sectors will be available to us. The ability to determine if a read is successful is also important and available from BIOS read routine (generally after a sufficient number of retries to rule out temporary faults like dust).

The routines with which we should be familiar are:

SELDSK—Select the disk drive given by register C for subsequent disk accesses, where register C contains 0 for drive A, 1 for drive B, etc.

SETTRK—Select the track number given by register C for subsequent disk accesses on the currently selected drive.

SETSEC—Select the sector number given by register C for subsequent disk accesses on the currently selected track of the currently selected drive.

READ—Read the disk sector currently selected. Upon return, register A will contain a 0 if no errors occurred and a 1 if a non-recoverable error condition occurred.

Based on the information obtained from the foregoing investigation, the following solution is suggested:

1. Use the BIOS primatives to find the bad sectors on a selected drive.
2. Use the BDOS primatives to create the directory entry for the '[unused].bad' file.
3. Determine the proper 1K block address of the bad sector and add it to the allocation map of '[unused].bad.'

While this does not entirely specify the program, it will provide a foundation.

Programming problem statement

To many, where to start the program coding is a piece of magic. We will approach our programming problem in the same way we approached the first half of this problem. First the original program request is restated: Read all the sectors on a diskette and group all unreadable sectors together under file name '[unused].bad.'

This is really two sections of programming requirements: Search for and keep track of all bad sectors on disk and build the directory entry containing this information.

Each of these programming tasks can be restated in terms of desired routines to perform specific tasks:

- Establish a link with the CP/M BIOS
- Discover which drive is to be checked
- Read all sectors and keep track of bad ones
- If no bad spots, say so and exit program

and

- Create the directory entry for '[unused].bad'
- Fix the allocation map to point to bad blocks
- Put the '[unused].bad' directory entry on the diskette
- State how many bad blocks were found
- Exit program

The entire program is contained in the 12 lines between 'org 100h' and 'jmp boot.' The rest of the code presented is in support of these few statements. Some of the subroutines called to perform tasks will call other subordinate routines. If any problems develop with the code, each module can be isolated and tested with minimal interference from other modules. This can shorten the debugging process by days.

Notice that the reading of the diskette to find bad spots is done at the BIOS level, while establishing the directory entry is done at the BDOS level. The higher level routines should always be used unless they will not cope with the particular problem that is being addressed.

When the number of bad blocks exceeds 16, another directory entry called '[unused1.].bad' is created to hold the overflow. Additional file names are created as long as there are bad blocks to isolate. If the disk has more than 32 bad blocks, perhaps it should be sent to the big disk drive in the sky for the rest it deserves.

If any problems develop. . .each module can be isolated and tested with minimal interference. . .

The routine 'findb' consists of three functions, since there are three logical sections of the disk and any or all may contain errors. The system tracks (0 and 1) may contain errors but are not a part of the data allocation map. Errors in the directory blocks (track 2 - blocks 0 and 1) cause any further testing to be meaningless, since an unreliable directory forces the diskette to be unusable. The remaining blocks (2 through 241) are a part of the allocation map and can be isolated if errors occur by including them in the directory entry '[unused].bad.'

Using the program

Before using this program to reclaim a diskette, it is recommended that the diskette be reformatted. If this is not possible, at least assure yourself that any existing files on the diskette do not contain unreadable sectors.

To use the program, insert both the disk containing the program 'findbad.com' and the diskette to be checked into the disk drives. It is possible that the diskette containing the program is the one to be checked. Assume that the program is on drive A and the suspected diskette is on drive B. In response to the CP/M prompt 'A>', type in 'findbad B:'. This will load the program 'findbad.com' from drive A and test the diskette on drive B for unreadable sectors. An alternative is to log-in on drive B with the command 'B:', and then execute the program from drive A with 'A:findbad'. This will load the program from drive A and test the currently logged drive (B). In any event, the only allowable parameter after the program name is a drive specification (of the form 'X:'). If no drive specification is given, the currently logged in drive is assumed to contain the diskette to check. This happens when the diskette to be checked also contains the program.

The program first checks the system tracks (0 and 1). If any read errors occur, the message '***warning*** system tracks bad***' will appear. Errors on the first two tracks prohibit the use of the diskette on drive A,

since all warm boots occur using the system tracks from the A drive. The diskette may be used in the B drive without harm.

Next, the program checks the first two data blocks. These blocks contain the directory for the diskette. If a read error occurs during this check, the message '***bad spot in directory—cannot continue***' is displayed, the program terminates, and control returns to CP/M. No other data blocks are checked since errors in the directory render the diskette useless.

Finally, all the remaining data blocks are checked. Any sectors which are unreadable cause the data block which contains them to be stored as a bad block. At the end of this phase of processing, the message 'XX bad blocks found' is displayed, where XX is replaced by the number of bad blocks or NO if no read errors occurred. If bad blocks occurred, the file name '[unused].bad' is created, the list of bad blocks is placed in the allocation map of the directory entry for '[unused].bad' and the file is closed.

If any bad blocks occurred in the preceding processing, by forcing their allocation to the file name '[unused.bad', they will no longer be available to CP/M for allocation and so are logically removed from the mainstream of processing.

Final comments

A word of caution is in order. This investigation was made with Tarbell disk controller and CP/M. I do not know whether or not the particulars of this solution can be carried across to other versions of CP/M BIOS. It would appear that at least the principles apply to the basic CP/M structure. □

DAMAGED DISK PROGRAM LISTING

```
                ;       FINDBAD WILL FIND BAD BLOCKS AND BUILD A FILE
                ;       NAMED [UNUSED].BAD TO CONTAIN THEM
                ;
0000 =          BOOT    EQU     0       ;CPM WARM BOOT
0005 =          BDOS    EQU     5       ;CPM BDOS ENTRY
0080 =          TBUFF   EQU     80H     ;COMMAND LINE BUFFER
                ;
004D =          TRACKS  EQU     77      ;NUMBER OF TRACKS PER DISK
001A =          SECTS   EQU     26      ;NUMBER OF SECTORS PER TRACK
0002 =          DBASE   EQU     2       ;TRACK BEGINNING DATA AREA
0002 =          BBASE   EQU     2       ;1ST BLOCK FOR DATA
00F1 =          MAXB    EQU     241     ;MAX NUMBER OF BLOCKS
0008 =          BLOCK   EQU     8       ;NUMBER OF SECTORS PER BLOCK
                ;
0100                    ORG     100H
0100 314F14     START   LXI     SP,DM+1000H
0103 CD2301             CALL    IBIOS   ;SET BIOS ENTRY & CHECK DRIVE
0106 CDA701             CALL    FINDB   ;ESTABLISH ALL BAD AREAS
0109 CA1801             JZ      NOBAD   ;SAY NO BAD SPOTS
010C CDDB02             CALL    OPENB   ;OPEN [UNUSED].BAD
010F CD1903             CALL    SETDM   ;FIX DM BYTES IN FCB
0112 CD6403             CALL    CLOSEB  ;CLOSE [UNUSED].BAD
0115 CD6D03             CALL    SETNUM  ;PUT NUMBER BAD BLOCKS IN MSG
0118 112D04     NOBAD   LXI     D,ENDMSG ;SAY HOW MANY BAD
011B 0E09       PMSG    MVI     C,9     ;PRINT BUFFER
011D CD0500             CALL    BDOS
0120 C30000             JMP     BOOT    ;RETURN TO CPM WITH WARM BOOT
                ;
                IBIOS
                ;   GET ACTUAL ADDRESSES OF BIOS ROUTINES
0123 2A0100             LHLD    BOOT+1  ;GET BASE ADDRESS OF BIOS VECTOR
0126 111B00             LXI     D,27    ;OFFSET TO SETTRK
0129 19                 DAD     D
012A 228A02             SHLD    SETTRK+1 ;FIX OUR CALL ADDRESS
012D 110300             LXI     D,3     ;OFFSET TO SETSEC
0130 19                 DAD     D
0131 228E02             SHLD    SETSEC+1 ;FIX OUR CALL ADDRESS
0134 110600             LXI     D,6     ;OFFSET TO DISK READ
0137 19                 DAD     D
0138 229102             SHLD    DREAD+1 ;FIX OUR CALL ADDRESS
                ;  CHECK FOR DRIVE SPECIFICATION
013B 3A8000             LDA     TBUFF   ;GET LENGTH OF COMMAND PARAMETERS
013E FE02               CPI     2       ;UNDER 2 = NO PARAMS
0140 D8                 RC
0141 CA6401             JZ      ERROR1  ;EXACTLY 2 IS ERROR
0144 2A8200             LHLD    TBUFF+2 ;AT LEAST THREE
0147 7C                 MOV     A,H     ;GET SHOULD BE ":"
0148 FE3A               CPI     ':'     ;CHECK IT
014A C26401             JNZ     ERROR1  ;ONLY ALLOW DRIVE SET
014D 7D                 MOV     A,L     ;NOW FOR DRIVE #
014E FE41               CPI     'A'     ;AT LEAST "A"?
0150 DA6401             JC      ERROR1
0153 FE45               CPI     'E'     ;NOT MORE THAN "D"?
0155 D26401             JNC     ERROR1
0158 E607               ANI     7       ;STRIP MOST BITS
015A 3D                 DCR     A       ;BACK OFF FOR 0 - 3
015B 5F                 MOV     E,A     ;DRIVE SPEC
015C 1600               MVI     D,0
015E 0E0E               MVI     C,14    ;SELECT DRIVE
0160 CD0500             CALL    BDOS
0163 C9                 RET
                ;
                ;  ERROR IN COMMAND LINE - ("FINDBAD" OR "FINDBAD B:")
0164 116A01     ERROR1  LXI     D,ERMSG1        ;SAY NO GO
0167 C31B01             JMP     PMSG    ;GO PRINT AND BOOT
                ;
016A 0D0A455252ERMSG1   DB      0DH,0AH,'ERROR IN COMMAND LINE'
0181 0D0A4D5553         DB      0DH,0AH,'MUST BE "FINDBAD" OR "FINDBAD X:"'
01A4 0D0A24             DB      0DH,0AH,'$'
                ;
                ;  ESTABLISH IF ANY DISK BLOCKS ARE BAD
01A7 CDC201     FINDB   CALL    CHKSYS  ;SEE IF ANY BAD BLOCKS ON TRACKS 0,1
01AA CD0302             CALL    CHKDIR  ;SEE IF ANY BAD BLOCKS IN DIRECTORY
01AD 0602               MVI     B,BBASE ;START AT FIRST DATA BLOCK
01AF CD5202     FINDBA  CALL    READB   ;READ THE BLOCK
01B2 C4C802             CNZ     SETBD   ;IF ERROR - ADD BLOCK TO LIST
01B5 04                 INR     B       ;BUMP TO NEXT BLOCK
01B6 78                 MOV     A,B     ;SEE IF ANY MORE TO CHECK
01B7 FEF1               CPI     MAXB    ;CHECK AGAINST MAX BLOCKS
01B9 DAAF01             JC      FINDBA
01BC 2A4B04             LHLD    DMCNT   ;GET NUMBER SECTORS BAD
01BF 7C                 MOV     A,H
01C0 B5                 ORA     L       ;SET ZERO FLAG IF NO BAD SPOTS
01C1 C9                 RET
                ;
                ;       SAY IF THERE ARE BAD SPOTS IN THE SYSTEM REGION
                                                        (TRACKS 0
01C2 210100     CHKSYS  LXI     H,1     ;H=TRACK 0, L=SECTOR 1
01C5 CD8202     CHKSY1  CALL    READS   ;GO READ A SECTOR
01C8 C2D201             JNZ     SYSERR  ;SAY BAD SPOT
01CB 7C                 MOV     A,H     ;MORE TO CHECK?
01CC FE02               CPI     2
01CE DAC501             JC      CHKSY1
01D1 C9                 RET
01D2 11DB01     SYSERR  LXI     D,ERMSG9
01D5 0E09               MVI     C,9     ;PRINT BUFFER
01D7 CD0500             CALL    BDOS
01DA C9                 RET
01DB 0D0A2A2A2AERMSG9   DB      0DH,0AH,'*** WARNING ***
                                                SYSTEM TRACKS BAD ***$'
                ;
                ;       SAY IF THERE ARE BAD SPOTS IN THE DIRECTORY BLOCKS
0203 0600       CHKDIR  MVI     B,0     ;START AT BLOCK ZERO
0205 CD5202     CHKDI1  CALL    READB   ;READ 1ST BLOCK
0208 C21302             JNZ     DIRERR  ;GO INDICATE PROBLEM IN DIR
020B 04                 INR     B       ;LOOK AT NEXT BLOCK
020C 78                 MOV     A,B     ;MORE DIRECTORY?
020D FE02               CPI     BBASE
020F DA0502             JC      CHKDI1
0212 C9                 RET             ;GO BACK IF DIR OK
0213 111902     DIRERR  LXI     D,ERMSG2
0216 C31B01             JMP     PMSG
0219 0D0A2A2A2AERMSG2   DB      0DH,0AH,'*** BAD SPOT IN DIRECTORY AREA - '
023C 43414E4E4F         DB      'CANNOT CONTINUE ***',0DH,0AH,'$'
                ;
                ;  READ ALL SECTORS IN BLOCK AND RETURN ZERO SET IF NO BAD
0252 CD6002     READB   CALL    CNVRTB  ;CONVERT TO TRACK/SECTOR IN H,L
0255 0E08               MVI     C,BLOCK ;NUMBER OF SECTORS PER BLOCK
0257 CD8202     READBA  CALL    READS   ;READ INTERLACED SECTOR
025A C0                 RNZ             ;ERROR IF NOT ZERO
025B 0D                 DCR     C       ;COUNT OFF SECTORS
025C C25702             JNZ     READBA
025F C9                 RET             ;RETURN ZERO FLAG SET=ALL OK
```

Continued on Page 130

Build your own microcomputer as you learn computer technology at home.

New from NRI! The Most Complete and Up-to-date Home Study Course Ever Offered

As the microprocessor revolutionizes the computer world and microcomputers appear almost everywhere, NRI brings you a new, convenient, and effective way to keep up with this expanding technology. It's NRI's Computer Technology Course, created and designed exclusively for learning at home in your spare time.

Featuring NRI's Exclusive Dual Language Microcomputer

NRI goes beyond book learning to give you practical, "hands-on" experience in designing circuitry, interfacing components, programming, and troubleshooting. As you learn, you actually assemble NRI's designed-for-learning microcomputer, incorporating the latest advances in the state of the art. It looks and operates like the finest of its kind, actually does more than many commercial units. But NRI engineers have designed components and planned assembly so it demonstrates important principles, gives you working experience in detecting and correcting problems. And it's yours to keep, put to work in your own home or business.

You also build and keep your own test instruments, including a transistorized volt-ohm meter and CMOS digital frequency counter. And NRI's Discovery Lab® broadens your horizons with specialized experiments and theory demonstrations.

The Proven Way to Learn at Home

You don't have to worry with travel, classes, or time lost from work when you learn the NRI way. As they have for more than 60 years of teaching technical subjects, NRI brings the material to you. You study in your spare time, at your convenience, using "bite-size" lessons that program material into logical segments for easier assimilation. You perform experiments and build equipment using kits we supply. And your personal NRI instructor is always available for consultation should you have questions or problems. Over a million students have already shown the effectiveness of NRI training.

Choice of Courses

Several courses are available, depending upon your needs and background. NRI's Master Course in Computer Technology starts with the fundamentals, explores basic electronics and digital theory, the total computer world, and the microcomputer. The Advanced Course, for students already versed in electronics and general computers, concentrates on the microprocessor and microcomputer. In both courses, you build all instruments and your own computer.

Send for Free Catalog... No Salesman Will Call

Get the details on these exciting new courses in NRI's free, 100-page catalog. Shows all kits and equipment, lesson outlines, and full information, including facts on other electronics courses. Mail the coupon today and we'll rush your catalog. No salesman will ever call. Keep up with the latest technology as you learn on your own computer. If coupon has been removed, write to NRI Schools, Computer Department, 3939 Wisconsin Ave., Washington, D.C. 20016.

NRI We Grow with You

NRI Schools
McGraw-Hill Continuing Education Center
3939 Wisconsin Avenue
Washington, D.C. 20016

NO SALESMAN WILL CALL
Please check for one free catalog only.

All career courses approved under GI Bill.
☐ Check for details.

- ☐ Computer Electronics Including Microcomputers
- ☐ TV/Audio/Video Systems Servicing
- ☐ Complete Communications Electronics with CB • FCC Licenses • Aircraft, Mobile, Marine Electronics
- ☐ CB Specialists Course
- ☐ Amateur Radio • Basic and Advanced
- ☐ Digital Electronics • Electronic Technology • Basic Electronics
- ☐ Small Engine Repair
- ☐ Electrical Appliance Servicing
- ☐ Automotive Mechanics
- ☐ Auto Air Conditioning
- ☐ Air Conditioning, Refrigeration, & Heating Including Solar Technology

Name (Please Print) Age

Street

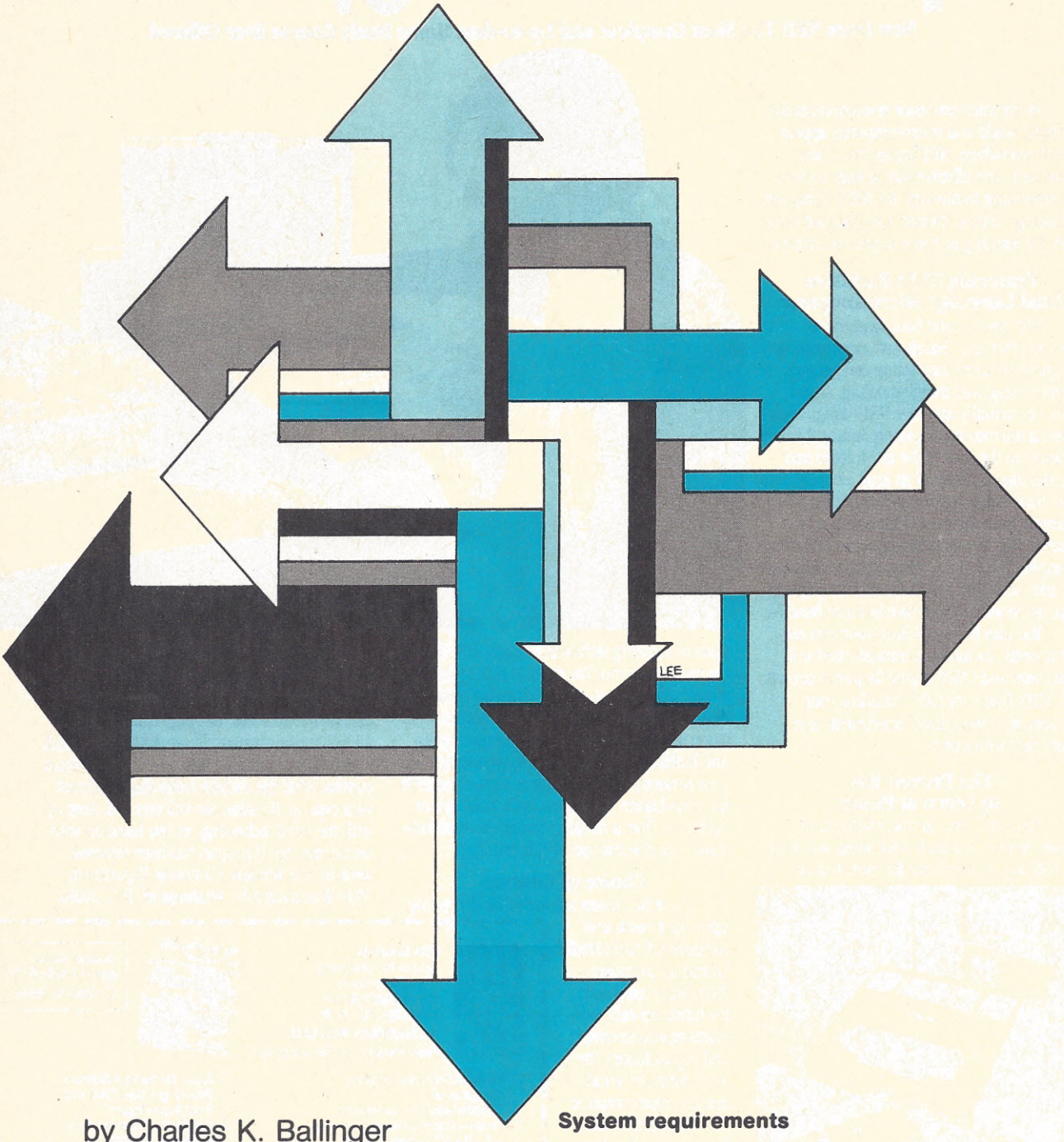
City/State/Zip

Accredited by the Accrediting Commission of the National Home Study Council

171-090

A User Compares:

HDOS or CP/M. . . Which Way to Go?



by Charles K. Ballinger

Those individuals out in the computer world running the Heath H-8 have long since discovered the disturbing lack of available software. One possible solution, however, has been found that might bring some relief. We refer to the CP/M operating system. It's useful, therefore, to examine the differences between HDOS and CP/M, taking aim at the advantages and disadvantages of both.

System requirements

HDOS	CP/M
H8 with at least 16K	H8 with at least 32K
H17 disk drive (1)	H17 disk unit (1)
terminal (CRT or hardcopy)	terminal (CRT or hardcopy)

Obviously CP/M's requirement of twice the core poses a serious deterrent for someone with only a 16 to 24K system.

The diskette storage capacity between HDOS and CP/M differs by a 2K margin. An HDOS disk of 102K capacity after the disk directory and overhead has 92K of usable space left; while the diskette formatted under CP/M after directory and overhead has 90K usable. These figures are for a disk that does not contain the operating system, i.e. the free space available on the second disk unit.

Both systems use identical boot-up procedures: you press 'o/rst', then REG, then PC, then the 'alter' key, alter the PC to address 030000 octal, press 'alter' again and then press 'go'. The disk will activate on drive A or 'syso' as appropriate, and the system will boot up. (Example 1 is the boot-up result from HDOS; example 2 is the same procedure from CP/M.)

```
ACTION? <BOOT> BOOT

SYSTEM HAS 36K OF RAM

VOLUME 241, MOUNTED ON SY0:
LABEL: ATOM20.BASIC.MAIN.DISK.3-18-79


HDOS VERSION 1.0
 ISSUE # 50.03.00
DATE (29-JUL-79)?
>
```

Example 1 — HDOS boot-up procedure

```
CP/M on Heath
32K Version 1.40
Copyright (C) 1979 Lifeboat Associates

A>SAVEUSER

Saveuser Vers 1.7 - for Heathkit Disk.
Saves 17 sectors starting at BIOS (2280H).
Copyright (C) 1979 Lifeboat Associates

Place CP/M SYSTEM DISK into drive A and
type RETURN to patch (or ^C to not patch)

User area patching completed.
A>
```

Example 2 — CP/M boot-up procedure

Generating the system

Both systems are remarkably similar in their approach to 'sysgening', the procedure that makes a working copy of the operating system so the original is not accidentally destroyed.

Of the two procedures, CP/M is far more difficult. You must read the manual twice before attempting it. The documentation provided is easy to follow and should present no problems. One note though: the word 'space' will appear in the front panel leds after pressing 'go'; this only holds true under an H8-4 serial card. When using the H8-5 card, the program will bypass this prompt to set the baud rate. (See example 3 for a run generating HDOS and example 4 of the CP/M generation.)

```
 SYSGEN
SYSGEN ISSUE # 50.03.00

DISMOUNTING ALL DISKS:

VOLUME 000  DISMOUNTED FROM SY0
 LABEL: HDOS 1.0 ISSUE #50.03.00 (COPYRIGHT(C) HEATH CO 1979)890-1-3

REMOVE THE DISK(S). HIT RETURN WHEN READY:

INSERT THE SYSTEM DISTRIBUTION SOURCE DISK. HIT RETURN WHEN READY:

INSERT DESTINATION:

INSERT SOURCE:

INSERT DESTINATION:
14 FILES COPIED
```

Example 3 — 'Sysgen' of HDOS

```
HOW MANY STOP BITS (1 OR 2) ?
TYPE "CR" FOR DEFAULT "1" 1

ARE YOU USING AN H8-4 IN ADDITION TO THE
H8-5 THAT THE CONSOLE IS ON (Y-N) ?
TYPE "CR" FOR DEFAULT "N" N

HOW MANY DISK DRIVES IN SYSTEM (1 OR 2) ?
TYPE "CR" FOR DEFAULT "2" 2

ENTER STEP RATE IN MSEC (6-40)
TYPE "CR" FOR DEFAULT "30" 12

CAN YOUR TERMINAL HANDLE LOWER CASE (Y/N) ?
TYPE "CR" FOR DEFAULT "Y" Y

PRESS "CR" AND CP/M WILL SIGN ON

CP/M on Heath
32K Version 1.40
Copyright (C) 1979 Lifeboat Associates

A>FTCOPY
FTCOPY BY BARRY A. WATZMAN
VERSION 1.2

THIS PROGRAM DUPLICATES A DISK,
DESTROYING ALL DATA PREVIOUSLY ON THE DESTINATION DISK

IS THAT WHAT YOU WANT (Y/N) ? Y

INSERT SOURCE DISK IN DRIVE A
AND DESTINATION DISK IN DRIVE B
THEN HIT RETURN

OPERATION COMPLETE.
TO COPY ADDITIONAL DISKETTES, TYPE "Y",
OTHERWISE HIT RETURN


INSERT BOOTABLE DISK IN DRIVE A
THEN C/R TO REBOOT


HOW MANY STOP BITS (1 OR 2) ?
TYPE "CR" FOR DEFAULT "1" 1

ARE YOU USING AN H8-4 IN ADDITION TO THE
H8-5 THAT THE CONSOLE IS ON (Y-N) ?
TYPE "CR" FOR DEFAULT "N" N

HOW MANY DISK DRIVES IN SYSTEM (1 OR 2) ?
TYPE "CR" FOR DEFAULT "2" 2

ENTER STEP RATE IN MSEC (6-40)
TYPE "CR" FOR DEFAULT "30" 12

CAN YOUR TERMINAL HANDLE LOWER CASE (Y/N) ?
TYPE "CR" FOR DEFAULT "Y" Y

PRESS "CR" AND CP/M WILL SIGN ON
```

Example 4 — CP/M generation

Of the two systems, HDOS by far has the best disk directory features from a user point of view. Since HDOS allows you to enter a date at the time you boot-up the system, that date is used whenever there is an entry in the directory. This is handy in order to check the date a file or program was entered.

CP/M directories treat the disk in an entirely different manner and take some getting use to. No file protection is allowed as in HDOS, so in several respects CP/M is not the system for an end-user unless he is well-trained on the system.

Feature Checklist

Category	HDOS	CP/M
Disk Basic	X	
Text editor	X	X
Assembler	X	X
Debugger for assembler	X	X
Sysgen	X	X
Pip	X	X
Onecopy	X	
Ftcopy		X
Flags	X	
Set	X	
Init17	X	
Test17	X	
Format		X

The utility features provided by both systems are similar in many cases, not only to the name of the program invoked, but to the format in which the information must be entered. While CP/M does not have the utilities to test the disk drives and the media; the Format program does initialize a disk for later use by CP/M. One convenient feature of the CP/M Ftcopy program is the ability to copy a disk and not have the requirement that the output disk be preformatted as is the case with a disk-to-disk copy under HDOS.

Pros and cons examined

HDOS: The only major drawback with HDOS is its inability to read a series of disks in the alternate drive under program control, since you must mount and dismount all volumes. While this may not seem important, it does prevent reading files that may be maintained on a weekly or monthly basis and then reading them all at once to compile statistics or yearly reports without going through a merge of some kind.

CP/M: A lot more potential lies in this system than in HDOS, since Heath will not release the information necessary for software houses or users to adapt it to their systems. Since the Heath is a hobbyist system, I expected the software to allow me to make changes or enhancements. I soon found that no source listings were allowed nor are any usable entry points given under the disk system. The large amount of software available for CP/M opens a whole field of capabilities. I recommend this system as an additional purchase to HDOS for this reason alone. Many may desire to purchase only CP/M, but I like the flexibility of running two operating systems. □

No. 15:
Take a byte.

Software for most popular 8080/Z80 computer disk systems including*
NORTH STAR, ICOM, MICROPOLIS, DYNABYTE DB8/2 & DB8/4, EXIDY SORCERER, SD SYSTEMS, ALTAIR, VECTOR MZ, MECA, 8" IBM, HEATH H17 & H89, HELIOS, IMSAI VDP42 & 44, REX, NYLAC, INTERTEC SUPER-BRAIN, VISTA V80 and V200, TRS-80* MODEL I and MODEL II, ALTOS, OHIO SCIENTIFIC, DIGI-LOG, KONTRON PSI-80, IMS 5000** *diskette formats and* ***CSSN BACKUP *cartridge tapes.*

Prices reflect distribution on 8" single density diskettes. If a format is requested which requires additional diskettes, a surcharge of $8. per additional diskette will be added. A surcharge of $25 will be added for software on CSSN format DC 300XL cartridges. Media charge for 5440 disk is $100.

CP/M® VERSION 2 FOR TRS-80 MODEL II NOW AVAILABLE

Genuine CP/M for Apple II coming soon! Call for details.

All Lifeboat programs require CP/M, unless otherwise stated

Software with Manual / Manual Alone

☐ **CP/M* FLOPPY DISK OPERATING SYSTEM** — Digital Research's operating system configured for many popular micro-computers and disk systems:

System	Version	Price
North Star Single Density	1.4	145/25 v
North Star Double Density	1.4	145/25
North Star Double/Quad	2.x	170/25
Durango F-85	2.x	170/25
iCOM Micro-Disk 2411	1.4	145/25
iCOM 3712	1.4	170/25 v*
iCOM 3812	1.4	170/25 *
Mits 3202/Altair 8800	1.4	145/25
Heath H8 + H17	1.4	145/25 m
Heath H89	1.4	145/25 m
Heath H89 by Magnolia	1.4	250/25◊
Heath H89 by Magnolia	2.x	300/25◊
Onyx C8001	2.x	300/25◊
TRS-80 Model I	1.4	145/25 m
TRS-80 Model II	2.x	170/25
TRS-80 Model II + Corvus	2.x	250/25
Processor Technology Helios II	1.4	145/25
Cromemco System 3	1.4	145/25
Intel MDS Single Density	1.4	145/25
Intel MDS Single Density	2.x	170/25
Micropolis Mod I	1.4	145/25 v
Micropolis Mod II	1.4	145/25 v

The following configurations are scheduled for release soon:

Apple II SoftCard	2.x	350/25◊
North Star Double/Quad + Corvus	2.x	250/25
North Star Horizon HD-1	2.x	250/25
Ohio Scientific C3	2.x	200/25
Ohio Scientific C3-C	2.x	250/25
Micropolis Mod II	2.x	200/25
Mostek MDX STD Bus System	2.x	350/25 **
iCOM 3812	2.x	225/25 *
iCOM 4511/Pertec D3000	2.x	375/25 * +

*Software consists of the operating system, text editor, assembler, debugger and other utilities for file management and system maintenance. Complete set of Digital Research's documentation and additional implementation notes included. Systems marked * and ** include firmware on 2708 and 2716. Systems marked + include 5440 media charge. Systems marked Ⓜ require the special Ⓜ versions of software in this catalog. Systems marked v have minor variants available to suit console interface of system. Call or write for full list of options. ◊ includes hardware addition to allow our standard versions of software to run under it.*

DIGITAL RESEARCH

☐ **MP/M** — Installed for single density MDS-800. Multi-processing derivative of the CP/M operating system. Manual includes CP/M2 documentation .... $300/$50

☐ **MAC-8080** — Macro assembler. Full Intel macro definitions. Pseudo Ops include RPC, IRP, REPT, TITLE, PAGE, and MACLIB. Produces absolute hex output plus symbol table file for use by SID and ZSID (see below) .... $120/$15

☐ **SID-8080** — Symbolic debugger. Full trace, pass count and breakpoint program testing. Has back-trace and histogram utilities. When used with MAC, provides full symbolic display of memory labels and equated values .... $105/$15

☐ **ZSID-Z80** — Symbolic debugger with all features of SID .... $130/$15

Back Again!

☐ **TEX** — Text output formatter to create paginated, page-numbered and justified copy. Output can be directed to printer or disk .... $105/$15

☐ **DESPOOL** — Utility program to permit simultaneous printing from text files while executing other programs .... $80/$10

☐ **Z80 DEVELOPMENT PACKAGE** — Consists of: (1) disk file line editor, with global inter and intra-line facilities; (2) Z80 relocating assembler, Zilog/Mostek mnemonics, conditional assembly and cross reference table capabilities; (3) linking loader producing absolute Intel hex disk file .... $95/$20

☐ **ZDT** — Z80 Monitor Debugger to break and examine registers with standard Zilog/Mostek mnemonic disassembly displays. $35 when ordered with Z80 Development Package .... $50/$10

AVOCET SYSTEMS

☐ **XASM-68** — Non-macro cross-assembler with nested conditionals and full range of pseudo operations. Assembles from standard Motorola MC6800 mnemonics to Intel hex .... $200/$25

☐ **XASM-65** — As XASM-68 for MOS Technology MCS-6500 series mnemonics .... $200/$25

New! ☐ **XASM-48** — As XASM-68 for Intel MCS-48 and UPI-41 families .... $200/$25

New! ☐ **XASM-18** — As XASM-68 for RCA 1802 .... $200/$25

☐ **DISTEL** — Disk based disassembler to Intel 8080 or TDL/Xitan Z80 source code, listing and cross reference files, Intel or TDL/Xitan pseudo ops optional. Runs on 8080 .... $65/$10

☐ **DISILOG** — As DISTEL to Zilog/Mostek mnemonic files. .... $65/$10

☐ **SMAL/80** Structured Macro Assembler Language — Package of powerful general purpose text macro processor and SMAL structured language compiler. SMAL is an assembler language with IF-THEN-ELSE, LOOP-REPEAT-WHILE, DO-END, BEGIN-END constructs .... $75/$15

New Address

Software with Manual / Manual Alone

☐ **tiny C** — Interactive interpretive system for teaching structured programming techniques. Manual includes full source listings .... $105/$50

☐ **BDS C COMPILER** — Supports most features of language, including Structures, Arrays, Pointers, recursive function evaluation, overlays. Includes linking loader, library manager, and library containing general purpose, file I/O, and floating point functions. Lacks initializers, statics, floats and longs. Documentation includes "The C PROGRAMMING LANGUAGE" by Kernighan and Ritchie .... $145/$25

New Features

☐ **WHITESMITHS C COMPILER** — The ultimate in systems software tools. Produces faster code than a pseudo-code Pascal with more extensive facilities. Conforms to the full UNIX* Version 7 C language, described by Kernighan and Ritchie, and makes available over 75 functions for performing I/O, string manipulation and storage allocation. Linkable to Microsoft REL files. Requires 60K CP/M .... $630/$30

MICROSOFT

☐ **BASIC-80** — Disk Extended BASIC, ANSI compatible with long variable names, WHILE/WEND, chaining, variable length file records .... $325/$25

☐ **BASIC COMPILER** — Language compatible with BASIC-80 and 3-10 times faster execution. Produces standard Microsoft relocatable binary output. Includes MACRO-80. Also linkable to FORTRAN-80 or COBOL-80 code modules .... $350/$25

☐ **FORTRAN-80** — ANSI 66 (except for COMPLEX) plus many extensions. Includes relocatable object compiler, linking loader, library with manager. Also includes MACRO-80 (see below) .... $425/$25

☐ **COBOL-80** — Level 1 ANSI '74 standard COBOL plus most of Level 2. Full sequential, relative, and indexed file support with variable file names. STRING, UNSTRING, COMPUTE, VARYING/UNTIL, EXTEND, CALL, COPY, SEARCH, 3-dimensional arrays, compound and abbreviated conditions, nested IF. Powerful interactive screen-handling extensions. Includes compatible assembler, linking loader, and relocatable library manager as described under MACRO-80 .... $700/$25

☐ **MACRO-80** — 8080/Z80 Macro Assembler. Intel and Zilog mnemonics supported. Relocatable linkable output. Loader, Library Manager and Cross Reference List utilities included .... $149/$15

☐ **XMACRO-86** — 8086 cross assembler. All Macro and utility features of MACRO-80 package. Mnemonics slightly modified from Intel ASM86. Compatibility data sheet available .... $275/$25

☐ **EDIT-80** — Very fast random access text editor for text with or without line numbers. Global and intra-line commands supported. File compare utility included. .... $89/$15

☐ **PASCAL/M*** — Compiles enhanced Standard Pascal to compressed efficient Pcode. Totally CP/M compatible. Random access files. Both 16 and 32-bit Integers. Runtime error recovery. Convenient STRINGs. OTHERWISE clause on CASE. Comprehensive manual (90 pp, indexed). SEGMENT provides overlay structure. INPORT, OUTPORT and untyped files for arbitrary I/O. Requires 56K CP/M. Specify 1) 8080 CP/M, 2) Z80 CP/M, or 3) Cromemco CDOS .$175/$20

New Features

☐ **PASCAL/Z** — Z80 native code PASCAL compiler. Produces optimized, ROMable re-entrant code. All interfacing to CP/M is through the support library. The package includes compiler, relocating assembler and linker, and source for all library modules. Variant records, strings and direct I/O are supported. Requires 56K CP/M .... $395/$25

☐ **PASCAL/MT** — Subset of standard PASCAL. Generates ROMable 8080 machine code. Symbolic debugger included. Supports interrupt procedures, CP/M file I/O and assembly language interface. Real variables can be BCD, software floating point, or AMD 9511 hardware floating point. Includes strings enumerations and record data types. Manual explains BASIC to PASCAL conversion. Requires 32K. .... $250/$30

New Version

New Features

☐ **ALGOL-60** — Powerful block-structured language compiler featuring economical run-time dynamic allocation of memory. Very compact (24K total RAM) system implementing almost all Algol 60 report features plus many powerful extensions including string handling direct disk address I/O etc. .... $199/$20

☐ **CBASIC-2** Disk Extended BASIC — Non-interactive BASIC with pseudo-code compiler and run-time interpreter. Supports full file control, chaining, integer and extended precision variables, etc. .... $120/$15

MICRO FOCUS

☐ **STANDARD CIS COBOL** — ANSI '74 COBOL standard compiler fully validated by U.S. Navy tests to ANSI level 1. Supports many features to level 2 including dynamic loading of COBOL modules and a full ISAM file facility. Also, program segmentation, interactive debug and powerful interactive extensions to support protected and unprotected CRT screen formatting from COBOL programs used with any dumb terminal .... $850/$50

☐ **FORMS 2** — CRT screen editor. Output is COBOL data descriptions for copying into CIS COBOL programs. Automatically creates a query and update program of indexed files using CRT protected and unprotected screen formats. No programming experience needed. Output program directly compiled by STANDARD CIS COBOL .... $200/$20

New Phone

Software with Manual / Manual Alone

Tasty lower prices!

EIDOS SYSTEMS

☐ **KISS** — Keyed Index Sequential Search. Offers complete Multi-Keyed Index Sequential and Direct Access file management. Includes built-in utility functions for 16 or 32 bit arithmetic, string/integer conversion and string compare. Delivered as a relocatable linkable module in Microsoft format for use with FORTRAN-80 or COBOL-80, etc. .... $335/$23

☐ **KBASIC** — Microsoft Disk Extended BASIC version 4.51 integrated by implementation of nine additional commands in language. Package includes KISS.REL as described above, and a sample mail list program. .... $585/$45

To licensed users of Microsoft BASIC-80 (MBASIC) .... $435/$45

New

☐ **XYBASIC** Interactive Process Control BASIC — Full disk BASIC features plus unique commands to handle byte rotate and shift and to test and set bits. Available in several versions:

Integer ROMable	$350/$25
Integer ROM squared	$350/$25
Integer CP/M	$350/$25
Extended ROMable	$450/$25
Extended ROM squared	$450/$25
Extended CP/M	$450/$25
Extended Disk CP/M	$550/$25
Integer CP/M Run Time Compiler	$350/$25
Extended CP/M Run Time Compiler	$450/$25

New!

☐ **RECLAIM** — A utility to validate media under CP/M. Program tests a diskette or hard disk surface for errors, reserving the imperfections in invisible files, and permitting continued usage of the remainder. Essential for any hard disk. Requires CP/M version 2. .... $80/$5

☐ **BASIC UTILITY DISK** — Consists of: (1) CRUNCH-14 — Compacting utility to reduce the size and increase the speed of programs in Microsoft BASIC 4.51, BASIC-80 and TRS-80 BASIC. (2) DPFUN — Double precision subroutines for computing nineteen transcendental functions including square root, natural log, log base 10, sine, arc sine, hyperbolic sine, hyperbolic arc sine, etc. Furnished in source on diskette and documentation .... $50/$35

☐ **STRING/80** — Character string handling plus routines for direct CP/M BDOS calls from FORTRAN and other compatible Microsoft languages. The utility library contains routines that enable programs to chain to a COM file, retrieve command line parameters, and search file directories with full wild card facilities. Supplied as linkable modules in Microsoft format. .... $95/$20

☐ **STRING/80** source code available separately $295/NA

☐ **THE STRING BIT** — FORTRAN character string handling. Routines to find, fill, pack, move, separate, concatenate and compare character strings. This package completely eliminates the problems associated with character string handling in FORTRAN. Supplied with source .... $65/$15

☐ **VSORT** — Versatile sort/merge system for fixed length records with fixed or variable length fields. VSORT can be used as a stand-alone package or loaded and called as a subroutine from CBASIC-2. When used as a subroutine, VSORT maximizes the use of buffer space by saving the TPA on disk and restoring it on completion of sorting. Records may be up to 255 bytes long with a maximum of 5 fields. Upper/lower case translation and numeric fields supported. .... $175/$20

☐ **CPM/374X** — Has full range of functions to create or re-name an IBM 3741 volume, display directory information and edit the data set contents. Provides full file transfer facilities between 3741 volume data sets and CP/M files .... $195/$10

☐ **BSTAM** — Utility to link one computer to another also equipped with BSTAM. Allows file transfers at full data speed (no conversion to hex), with CRC block control check for very reliable error detection and automatic retry. We use it! It's great! Full wildcard expansion to send ★.COM, etc. 9600 baud with wire. 300 baud with phone connection. **Both** ends need one. Standard and Ⓜ versions can talk to one another. .... $150/$10

☐ **WHATSIT?*** Interactive data-base system using associative tags to retrieve information by subject. Hashing and random access used for fast response. Requires CBASIC-2 .... $175/$25

☐ **SELECTOR III-C2** — Data Base Processor to create and maintain multi Key data bases. Prints formatted sorted reports with numerical summaries or mailing labels. Comes with sample applications, including Sales Activity, Inventory, Payables, Receivables, Check Register, and Client/Patient Appointments, etc. Requires CBASIC-2. Supplied in source .... $295/$20

☐ **GLECTOR** — General Ledger option to SELECTOR III-C2. Interactive system provides for customized COA. Unique chart of transaction types insure proper double entry bookkeeping. Generates balance sheets, P&L statements and journals. Two year record allows for statement of changes in financial position report. Supplied in source. Requires SELECTOR III-C2, CBASIC-2 and 56K system .... $350/$25

Prices and specifications subject to change without notice.

Software with Manual / Manual Alone

☐ **CBS** — Configurable Business System is a comprehensive set of programs for defining custom data files and application systems without using a programming language such as BASIC, FORTRAN, etc. Multiple key fields for each data file are supported. Set-up program customizes system to user's CRT and printer. Provides fast and easy interactive data entry and retrieval **with transaction processing.** Report generator program does complex calculations with stored and derived data, record selection with multiple criteria, and custom formats. Sample inventory and mailing list systems included. **No support language required** .... $295/$40

MICRO DATA BASE SYSTEMS

☐ **HDBS** — Hierarchical Data Base System. CODASYL oriented with FILEs, SETs, RECORDs and ITEMs which are all user defined. ADD, DELETE, UPDATE, SEARCH, and TRAVERSE commands supported. SET ordering is sorted, FIFO, LIFO, next or prior. One to many set relationship supported. Read/write protection at the FILE level. Supports FILEs which extend over multiple floppy or hard disk devices.

☐ **MDBS** — Micro Data Base System. Full network data base with all features of HDBS plus multi-level read/write protection for FILE, SET, RECORD and ITEM. Explicit representation of one to one, one to many, many to many, and many to one SET relationships. Supports multiple owner and multiple record types within SETs. HDBS files are fully compatible.

☐ **HDBS-Z80** version .... $250/$40**

☐ **MDBS-Z80** version .... $750/$40**
8080 version available at $75 extra.

When ordering, specify one of the language interfaces listed below. Additional language interfaces available at time of purchase for $100 or $125 if purchased later.

**The single manual covering HDBS and MDBS when purchased alone comes without specific language interface manual. Manuals are available for the following Microsoft languages:

1) MBASIC 4.51, 2) BASIC-80 5.0, 3) Compiled BASIC or FORTRAN-80, 4) COBOL-80, 5) MACRO-80, ... $NA/$10

MICROPRO

☐ **SUPER-SORT I** — Sort, merge, extract utility as absolute executable program or linkable module in Microsoft format. Sorts fixed or variable records with data in binary, BCD, Packed Decimal, EBCDIC, ASCII, floating & fixed point, exponential, field justified, etc. Even variable number of fields per record! .$225/$25

☐ **SUPER-SORT II** — Above available as absolute program only .... $175/$25

☐ **SUPER-SORT III** — As II without SELECT/EXCLUDE .... $125/$25

New!

☐ **DATASTAR** — Professional forms control entry and display system for key-to-disk data capture. Menu driven with built-in learning aids. Input field verification by length, mask, attribute (i.e. upper case, lower case, numeric, auto-dup, etc.). Built-in arithmetic capabilities using keyed data, constant and derived values. Visual feedback for ease of forms design. Files compatible with CP/M-MP/M supported languages. Requires 32K CP/M .... $350/$35

☐ **WORD-STAR** — Menu driven visual word processing system for use with standard terminals. Text formatting performed on screen. Facilities for text paginate, page number, justify, center and underscore. User can print one document while simultaneously editing a second. Edit facilities include global search and replace, Read/Write to other text files, block move, etc. Requires CRT terminal with addressable cursor positioning .... $445/$40

☐ **WORD-STAR Customization Notes** — For sophisticated users who do not have one of the many standard terminal or printer configurations in the distribution version of WORD-STAR .... NA/$95

☐ **WORD-MASTER** Text Editor — In one mode has superset of CP/M's ED commands including global searching and replacing, forwards and backwards in file in video mode, provides full screen editor for users with serial addressable-cursor terminal .... $145/$25

☐ **TEXTWRITER III** — Text formatter to justify and paginate letters and other documents. Special features include insertion of text during execution from other disk files or console, permitting recipe documents to be created from linked fragments on other files. Has facilities for sorted index, table of contents and footnote insertions. Ideal for contracts, manuals, etc. Now compatible with Electric Pencil* prepared files. .... $125/$20

Now applications software for Microsoft's BASIC interpreter!

PEACHTREE SOFTWARE

☐ **GENERAL LEDGER** — Records details of all financial transactions. Generates a balance sheet and an income statement. Flexible and adaptable design for both small businesses and firms performing client writeup services. Produces reports as follows: Trial Balance, Transaction Registers, Balance Sheet, Prior Year Comparative Balance Sheet, Income Statement, Prior Year Comparative Income Statement and Department Income Statements. Interactive with other PEACHTREE accounting packages. Supplied in source code for Microsoft BASIC .... $990/$30

Software with Manual / Manual Alone

☐ **ACCOUNTS PAYABLE** — Tracks current and aged payables and incorporates a check writing feature. Maintains a complete vendor file with information on purchase orders and discount terms as well as active account status. Produces reports as follows: Open Voucher Report, Accounts Payable Aging Report and Cash Requirements. Provides input to PEACHTREE General Ledger. Supplied in source code for Microsoft BASIC ..... **$990/$30**

☐ **ACCOUNTS RECEIVABLE** — Generates invoice register and complete monthly statements. Tracks current and aged receivables. Maintains customer file including credit information and account status. The current status of any customer account is instantly available. Produces reports as follows: Aged Accounts Receivable, Invoice Register, Payment and Adjustment Register and Customer Account Status Report. Provides input to PEACHTREE General Ledger. Supplied in source code for Microsoft BASIC .. **$990/$30**

☐ **PAYROLL** — Prepares payroll for hourly, salaried and commissioned employees. Generates monthly, quarterly and annual returns. Prepares employee W-2's. Includes tables for federal withholding and FICA as well as withholding for all 50 states plus up to 20 cities from pre-computed or user generated tables. Will print checks, Payroll Register, Monthly Summary and Unemployment Tax Report. Provides input to PEACHTREE General Ledger. Supplied in source code for Microsoft BASIC ..... **$990/$30**

☐ **INVENTORY** — Maintains detailed information on each inventory item including part number, description, unit of measure, vendor and reorder data, item activity and complete information on current item costs, pricing and sales. Produces reports as follows: Physical Inventory Worksheet, Inventory Price List, Departmental Summary Report, Inventory Status Report, The Reorder Report and the Period-to-Date and Year-to-Date reports. Supplied in source code for Microsoft BASIC ..... **$1190/$30**

GRAHAM-DORIAN SOFTWARE SYSTEMS

☐ Comprehensive accounting software written in CBASIC-2 and supplied in source code. Each software package can be used as a stand-alone system or integrated with the General Ledger for automatic posting to ledger accounts. Requires CBASIC-2.

GENERAL LEDGER	$995/$35
ACCOUNTS PAYABLE	$995/$35
ACCOUNTS RECEIVABLE	$995/$35
INVENTORY SYSTEM	$590/$35
JOB COSTING	$995/$35
APARTMENT MANAGEMENT	$590/$35
CASH REGISTER	$590/$35

☐ **POSTMASTER** — A comprehensive package for mail list maintenance that is completely menu driven. Features include keyed record extraction and label production. A form letter program is included which provides neat letters on single sheet or continuous forms. Compatible with NAD files. Requires CBASIC-2 ..... **$150/$15**

STRUCTURED SYSTEMS GROUP

☐ **GENERAL LEDGER** — Interactive and flexible system providing proof and report outputs. Customization of COA created interactively. Multiple branch accounting centers. Extensive checking performed at data entry for proof, COA correctness, etc. Journal entries may be batched prior to posting. Closing procedure automatically backs up input files. Now includes Statement of Changes in Financial Position. Requires CBASIC-2 ..... **$1250/$25**

☐ **ACCOUNTS RECEIVABLE** — Open item system with output for internal aged reports and customer-oriented statement and billing purposes. On-Line Enquiry permits information for Customer Service and Credit departments. Interface to General Ledger provided if both systems used. Requires CBASIC-2. ..... **$1250/$25**

☐ **ACCOUNTS PAYABLE** — Provides aged statements of accounts by vendor with check writing for selected invoices. Can be used alone or with General Ledger and/or with NAD. Requires CBASIC-2 ..... **$1250/$25**

☐ **PAYROLL** — Flexible payroll system handles weekly, bi-weekly, semi-monthly and monthly payroll periods. Tips, bonuses, reimbursements, advances, sick pay, vacation pay, and compensation time are all part of the payroll records. Prints government required periodic reports and will post to multiple SSG General Ledger accounts. Requires CBASIC-2 and 54K of memory ..... **$1250/$25**

☐ **INVENTORY CONTROL SYSTEM** — Performs control functions of adding and depleting stock items, adding new items and deleting old items. Tracks quantity of items on hand, on order and back-ordered. Optional hard copy audit trail is available. Reports include Master Item List, Stock Activity, Stock Valuation and Re-order List. Requires CBASIC-2 **$1250/$25**

☐ **ANALYST** — Customized data entry and reporting system. User specifies up to 75 data items per record. Interactive data entry, retrieval, and update facility makes information management easy. Sophisticated report generator provides customized reports using selected records with multiple level break-points for summarization. Requires a disk sort utility such as QSORT, SUPER-SORT or VSORT and CBASIC-2. ..... **$250/$15**

☐ **LETTERIGHT** — Program to create, edit and type letters or other documents. Has facilities to enter, display, delete and move text, with good video screen presentation. Designed to integrate with NAD for form letter mailings. Requires CBASIC-2 ..**$200/$25**

☐ **NAD** Name and Address selection system — Interactive mail list creation and maintenance program with output as full reports with reference data or restricted information for mail labels. Transfer system for extraction and transfer of selected records to create new files. Requires CBASIC-2 ..... **$100/$20**

☐ **QSORT** — Fast sort/merge program for files with fixed record length, variable field length information. Up to five ascending or descending keys. Full back-up of input files created ..... **$100/$20**

★★★★★★★★ CONDIMENTS

☐ **HEAD CLEANING DISKETTE**—Cleans the drive Read/Write head in 30 seconds. Diskette absorbs loose oxide particles, fingerprints, and other foreign particles that might hinder the performance of the drive head. Lasts at least 3 months with daily use. Specify 5" or 8".
Single sided ..... **$20 each/$55 for 3**
Double sided ..... **$25 each/$65 for 3**

☐ **FLIPPY DISK KIT** — Template and instructions to modify single sided 5¼" diskettes for use of second side in single sided drives **$12.50**

☐ **FLOPPY SAVER** — Protection for center holes of 5" and 8" floppy disks. Only 1 needed per diskette. Kit contains centering post, pressure tool and tough 7 mil mylar reinforcing rings for 25 diskettes.
5", Kit **$14.95**
5", Rings only **$7.95**
8", Kit **$16.95**
8", Rings only **$8.95**

☐ **PASCAL USER MANUAL AND REPORT** — By Jensen and Wirth. The standard textbook on the language. Recommended for use by Pascal/Z, Pascal/M and Pascal/MT users **$12**

☐ **THE C PROGRAMMING LANGUAGE** — By Kernighan and Ritchie. The standard textbook on the language. Recommended for use by BDS C, tiny C, and Whitesmiths C users **$12**

☐ **STRUCTURED MICROPROCESSOR PROGRAMMING** — By the authors of SMAL/80. Covers structured programming, the 8080/8085 instruction set and the SMAL/80 language **$20**

☐ **ACCOUNTS PAYABLE & ACCOUNTS RECEIVABLE—CBASIC** — By Osborne/McGraw-Hill **$20**

☐ **GENERAL LEDGER—CBASIC**—By Osborne/McGraw-Hill **$20**

☐ **LIFEBOAT DISK COPYING SERVICE** — Transfer data or programs from one media format to another at a moderate cost ..... **from $25**

★★★★★★★★ Hearty Appetite.

*CP/M and MP/M are trademarks of Digital Research.
Z80 is a trademark of Zilog, Inc.
UNIX is a trademark of Bell Laboratories.
WHATSIT? is a trademark of Computer Headware.
Electric Pencil is a trademark of Michael Shrayer Software.
TRS-80 is a trademark of Tandy Corp.
Pascal/M is a trademark of Sorcim.

†Recommended system configuration consists of 48K CP/M, 2 full size disk drives, 24 x 80 CRT and 132 column printer.

Ⓜ Modified version available for use with CP/M as implemented on Heath and TRS-80 Model I computers.

Ⓛ User license agreement for this product must be signed and returned to Lifeboat Associates before shipment may be made.

ⓘⓧ This product Includes/eXcludes the language manual recommended in Condiments.

Ⓢ Serial number of CP/M system must be supplied with orders.
② Requires Z80 CPU.

Ordering Information

MEDIA FORMAT ORDERING CODES
When ordering, please specify format code.

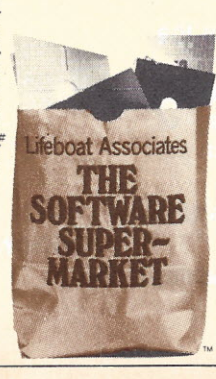
LIFEBOAT ASSOCIATES MEDIA FORMATS LIST
Diskette, cartridge disk and cartridge tape format codes to be specified when ordering software for listed computer or disk systems. All software products have specific requirements in terms of hardware or software support, such a MPU type, memory size, support operating system or language.

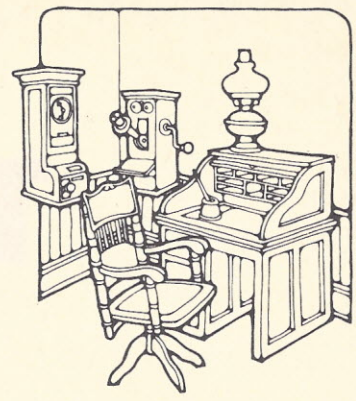
Computer system	Format Code
Altair 8800 Disk	See MITS 3200
Altos	A1*
Apple + Microsoft SoftCard	RG
BASF System 7100	RD
Blackhawk Single Density	Q3
Blackhawk Micropolis Mod II	Q2
CDS Versatile 3B	Q1
CDS Versatile 4	Q2
COMPAL-80	Q2
Cromemco System 3	A1*
Cromemco Z2D	R6
CSSN BACKUP (tape)	T1#
Delta	A1*
Digi-Log Microterm II	RD
Digital Microsystems	A1*
Discus	See Morrow Discus
Durango F-85	RL
Dynabyte DB8/2	R1
Dynabyte DB8/4	A1*
Exidy Sorcerer + Lifeboat CP/M	Q2
Exidy Sorcerer + Exidy CP/M	Q4
Heath H8 + H17/HZ2	P4
Heath 89 + Lifeboat CP/M	P4
Heath 89 + Magnolia CP/M	P7
Helios II	See Processor Technology
Horizon	See North Star
ICOM 2411 Micro Floppy	R3
ICOM 3712	A1
ICOM 3812	A1*
ICOM 4511 5440 Cartridge CP/M 1.4	D1#
ICOM 4511 5440 Cartridge CP/M 2.2	D2#
IMS 5000	RA
IMS 8000	A1*
IMSAI VDP-40	R4**
IMSAI VDP-42	R4**
IMSAI VDP-44	R5**
IMSAI VDP-80	A1**
Intecolor	See ISC Intecolor
Intel MDS Single Density	A1
Intertec SuperBrain DOS 0.1	R7
Intertec SuperBrain DOS 0.5-2.X	RJ
Intertec SuperBrain DOS 3.X	RK
ISC Intecolor 8063/8360/8963	A1
Kontron PSI-80	RF
Meca 5¼"	P6
Micromation (Except TRS-80 below)	A1*
Micropolis Mod I	Q1
Micropolis Mod II	Q2
MITS 3200/3202	B1
Morrow Discus	A1*
Mostek	A1
MSD 5¼"	RC
North Star Single Density	P1
North Star Double/Quad	P2
Nylac Single Density	Q3
Nylac Micropolis Mod. II	Q2
Ohio Scientific C3	A3
Onyx C8001	T2#
Pertec PCC 2000	A1*
Processor Technology Helios II	B2
RAIR Single Density	R9
RAIR Double Density	RE
Research Machines 8"	A1
Research Machines 5¼"	RH
REX	Q3
SD Systems 8"	A1*
SD Systems 5¼"	R3
Sorcerer	See Exidy Sorcerer
Spacebyte	A1
SuperBrain	See Intertec
Tarbell	A1*
TEI 5¼"	R3
TEI 8"	A1*
Thinkertoys	See Morrow Discus
TRS-80 Model I 5¼"	R2
TRS-80 Model I + FEC Freedom	RN
TRS-80 Model I + Micromation	A4*
TRS-80 Model I + Omikron 5¼"	RM
TRS-80 Model I + Omikron 8"	A1
TRS-80 Model I + Shuffleboard 8"	A1
TRS-80 Model II	A1*
VDP-40/42/44/80	See IMSAI
Vector MZ	Q2
Versatile	See CDS Versatile
Vista V80 5¼" Single Density	P5
Vista V200 5¼" Double Density	P6
Zenith Z89 + Lifeboat CP/M	P4
Zenith Z89 + Magnolia CP/M	P7

*Single-Side Single-Density disks are supplied for use with Double-Density and Double-Side 8" soft sector format systems.
**IMSAI formats are single density with directory offset of zero.
#A media surcharge of $25 for orders on tape formats T1 and T2 and of $100 for orders on disk formats D1 and D2 will be added.
The list of available formats is subject to change without notice. In case of uncertainty, call to confirm the format code for any particular equipment.

Prices F.O.B. New York. Shipping, handling and C.O.D. charges extra.
Manual cost applicable against price of subsequent software purchase.
The sale of each proprietary software package conveys a license for use on one system only.

VISA master charge





Articles Wanted...

Especially those dealing with novel or particularly useful applications or technologies aimed at the small businessman or modern educator. Payment ranges from $20 to $50 per published page, or may carry a company's byline in lieu of payment. Manuscripts should be typed, double spaced. Minimum length is four typewritten pages. Photos should be numbered and captioned; tables, listings and programs should be on separate pages, and camera-ready, i.e., suitable for reproduction. The publisher assumes no responsibility for artwork, photos or manuscripts. No acknowledgement is made unless accompanied with a self-addressed stamped envelope. For additional details contact: Interface Age, 16704 Marquardt Ave., Cerritos, CA 90701.

A Text Editor for the 6800

by Robert Hudson

A text editor is an important basic tool used to prepare the many types of files for everyday computer operations. It is a software program that allows one to easily create or modify text material. The text can include programs in Basic, Fortran, PL/1, or assembler language as well as job control language, data file, or magazine articles. The following text editor should be appealing to 6800 microcomputer owners due to its small size and powerful set of commands.

The program

The actual program requires memory from 0100 hex to 04FD hex. Also, memory is required on page zero for two buffers of 72 characters each and variable storage of 22 bytes. The variables have been arranged to facilitate the printout of the text buffer size and limits; therefore, do not rearrange 'topbuf,' 'cpntr,' or 'endbuf.' The program is begun by executing 0100 hex (cold start). A hot start is provided by beginning execution at 0103 hex. The cold start automatically erases the text buffer. A jump table contains system links to 'mikbug' type routines (except for one). As can be seen in the listing, the command table follows the jump table. The system messages have been arranged after the command table. The actual software begins at 01AE hex. The other jump to a 'mikbug' routine is found in the 'bottom' routine where a call is made to the OUT4HS subroutine. This routine outputs a double byte word followed by a space and it increments the index register twice. The location of this call is 029E hex. The stack is initialized to A060 hex in location 01F8 hex.

Text editor commands

TOP — The pointer is set to the beginning of the text buffer.

BOTTOM — The pointer is set to the end of the text buffer. Also an output is produced of three double byte hex numbers such as: 04FE 165C 43FE.

04FE indicates the beginning of the text buffer.

165C indicates the end of the current text in the buffer.

43FF indicates the current end or upper limit of the buffer.

These three numbers are produced whenever a 'bottom' command is executed.

APPEND — 'Append' was implemented to tack information onto the end of the present text material. The program assumes the appended data is a cassette file. An automatic 'bottom' is executed and control falls through to the cassette load routine.

NEXT nnn — This command moves the pointer through the text buffer. The number may be preceded by a minus sign (–) which causes the pointer to go back that many lines.

PRINT nnn — Prints nnn lines from the text buffer. The output begins from the line pointer and continues for nnn lines. The line pointer is not changed.

ERASE — This command clears the defined text buffer of any text which has been entered. All pointers are reset and can be checked by using the 'bottom' command. The size of the text buffer can be modified by the 'quantity' command.

DELETE nnn — Beginning at the current position of the line pointer, nnn lines are deleted from the text buffer.

INSERT — This command is used to enter text into the text buffer and can be terminated in several ways. The normal way to end an insert operation would be by pressing the ESC button. However, if the limits of the input buffer are exceeded, an error message will be printed and the program will revert back to the command mode. Also the backspace (5F hex) and line cancel (18 hex) are supported. The control characters can be modified to conform to individual systems.

FIND/string/ — This command will begin from the current line pointer, searching the text buffer for the string contained between the two de-

limiters (which may be any characters). If the string is not found, an error message will print. If the string is found, the line containing the string will print and the line pointer will be positioned at the beginning of the line. The 'R' command can be used to repeat this command. This means the 'find' argument does not have to be retyped to obtain a list of all occurrences of a given string.

CHANGE/string1/string2/

This command searches the text buffer from the current line pointer position looking for the first occurrence of 'string 1.' If 'string1' is found then 'string2' is inserted to take the place of 'string1.' If 'string1' is not found, an error message is printed and the line pointer remains at the current location. If 'string1' is found, the change is made and the line is printed and the current line pointer is changed to point to the beginning of the changed line. Also the 'R' (repeat) command can be used to execute a repeat change command. This allows one to change all occurrences of 'string1' to 'string2' easily. The two strings are defined by delimiters of any character, but the delimiter must be the same character for that operation.

QUANTITY nnn

This command allows one to respecify the size of the text buffer. For each nnn, the text buffer is increased by 256 bytes of memory. The new memory area is cleared beginning from the original buffer end location. This command can be executed while text is present in the text buffer. By using the 'bottom' command with its associated printout of the buffer limits, the text buffer size can be adjusted as required.

LOAD — This command will load a file from cassette. The data read in from the cassette is stored beginning at the location pointed to by the current line pointer. The software could be changed to do an automatic 'top' command. It was decided not to do an automatic 'top' command to retain the power and flexibility of entering data at any convenient setting of the line pointer. The tape (or disk) load and save routine will be dependent upon the individual system. These routines have been placed at the end of the software in an attempt to minimize the rewrite impact of adding your own load and save routines.

SAVE — This command will save data from the text buffer starting at the location pointed to by the line pointer to the end of the text stored in the buffer. Again an automatic 'top' was deliberately not performed for added power and flexibility. Also this routine, as for 'load,' is system dependent. Both of these routines must be rewritten for your system.

MOVE — This command will store the line pointed to by the line pointer in the move buffer on page zero. Before the actual store is accomplished, the length of the line is determined. If the length is greater than 72 characters, the line will not be moved into the move buffer but instead an error message will print. If the move was successful, the prompt character will print and the line to move is stored. The line pointer is still pointing to the beginning of the original line that was duplicated in the move buffer. Therefore, by typing a D, the original line will be deleted. By using the 'next' command, the line pointer can be changed to point to a different line, specifically the line where you want to move the original line. Once the new location of the line has been determined, use the 'here' command to insert the line stored in the move buffer into the text buffer. Repeated 'here' commands will duplicate the line stored in the move buffer.

HERE — This is used with the move command to store the current line in the move buffer to the text buffer.

R — A single character command will repeat the previous command in the command buffer. It is used to obtain a repeat 'find' or 'change' command. (All other commands can be complete words or leading character, but the repeat command can only be 'R'.)

X (monitor) — This command will jump to your system monitor. To reenter the text editor without erasing the text buffer, begin execution at 0103 hex (hot start). This command is not implemented in the listing. To implement, enter 58 hex at location 013C hex followed by the location in hex of the monitor's entry point. In the assembler listing, this code would go where the 'spare' label is located.

General command syntax comments

The number argument is optional and, if not entered, the program assumes number one. Also the maximum number value is 255. All commands can be shortened to the first character and may be entered in either upper or lower case. The 'repeat' command is an exception and must be entered as a single upper case character (i.e. 'R').

The assembler listing is for a non-mikbug system. The following changes must be made for a mikbug system:

Location	Code	
0106	7E E1 AC	JMP INPUT
0109	7E E1 D1	JMP OUTPUT
010C	7E E0 7E	JMP PDATA
029E	BD E0 C8	JSR OUT4HS

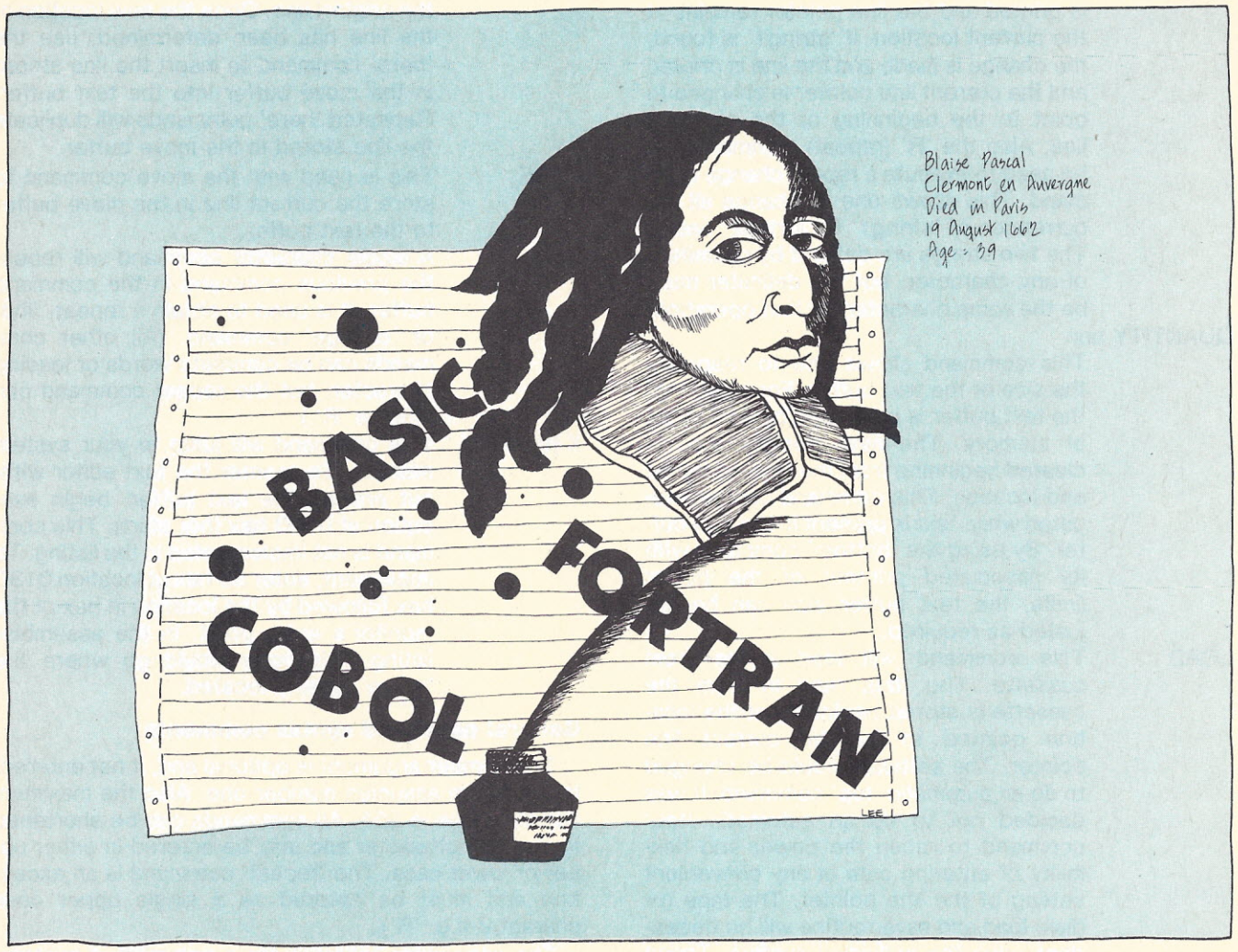
The break routine at 027D hex is written for an ACIA port. If a different type of port (PIA) is utilized, the break routine must be changed to accommodate it. □

Program on Page 132

Software Review

Pascal for CP/M
Digital Marketing's Pascal/M

by Alan R. Miller



Basic was the first, high-level language to be implemented on microcomputers. Two other high-level languages, Fortran and Cobol, have been available on larger computers for a much longer time. Both are now available on micros. These three languages continue to be popular since there are large amounts of software generally available. However, each has serious disadvantages.

Pascal is a high-level computer language that addresses many of the shortcomings of Basic, Fortran and Cobol. Several versions are available on micros.

Pascal: the inside story

The names of the common computer languages are acronyms.

Basic	Beginner's all-purpose symbolic instruction code
Fortran	Formula translation
Cobol	Common business-oriented language
Algol	Algorithmic language
APL	A programming language

But Pascal is named after the mathematician Blaise Pascal. The language was developed from Algol by Niklaus Wirth.

Some programming details of Pascal are similar to Basic, but in Pascal, a clear distinction is made between an assignment and a logical comparison. For example:

```
Pascal          Basic          Fortran
A : = 1         A = 1          A = 1
IF A = B        IF A = B       IF (A.EQ.B)
```

Basic, on the other hand, uses the same form for both constructions. Pascal also makes a distinction between array and function arguments. Square brackets are used for the former and parentheses for the latter.

```
Pascal          Basic          Fortran
B[3] : = SQRT(C)  B(3) = SQR(C)  B(3) = SQRT(C)
```

A carriage return signals the end of a statement in Basic and Fortran. Some Basics allow more than one statement on a line if the statements are separated by a colon or backslash. But even in this case, a carriage return terminates the last statement in the line.

CBasic allows a single statement to be spread over several lines. A backslash is placed at the end of a line to indicate that it is to be continued. In Microsoft Basic, a line can be continued by typing a line feed rather than a carriage return.

Pascal, by contrast, uses a free format: a semicolon separates one statement from the next. Carriage returns are ignored. Thus, if there is one Pascal statement on a line, there is a semicolon at the end of the line.

```
A : = 5;
```

If there are two or three statements on a line, a semicolon is used at the end of each statement

```
A : =5; B : = 3; C : = A;
```

including the last one. Alternately, one statement can be spread over several lines since carriage returns are ignored. Thus, long statements can be arranged for easy reading:

```
IF A < B
  THEN C : = 5
  ELSE C : = 8;
```

Program layout

Pascal programs begin with the program name and usually the names of the required input and output devices. For example, the statement:

```
PROGRAM curfit( input, output);
```

declares the program name to be Curfit and specifies that both console input and output will be needed. The line terminates with the usual semicolon.

The next lines should be comments describing what the program does and how it is used. Comments in Pascal are bracketed with a pair of braces or with a combination of parentheses and asterisks:

```
{ This is a Pascal comment }
(* This is also a Pascal comment *)
```

The comment is not a Pascal statement and so a final semicolon is not needed. Comments can be spread over several lines since the carriage return is ignored, although some people think this is poor programming practice.

```
(* This is a
   multiline
   comment *)
```

Comments can also be placed on the same line as regular statements. In fact, comments can be inserted into the middle of a statement. For example:

```
PROGRAM (* to *) solve (* linear matrices *)
(output);
```

is equivalent to:

```
PROGRAM solve (output);
```

Constants and variables are declared in the next part of the Pascal program. All such identifiers must be declared prior to use. Furthermore, constants are constant. They cannot be changed after they are declared. Identifiers begin with a letter and may continue with letters, digits and special characters. For example:

```
SUM_X_SQUARED     and
File#5
```

are valid identifiers. Names may be as long as 80 characters, but only the first eight are significant. Thus:

```
Sum_X_squared
Sum_X             and
Sum_Y_squared
```

are all different names.

The constant declaration section begins with the word 'const.' Then the definitions follow. Standard Pascal allows variables of type 'real', 'integer', 'char' (i.e., one Ascii character), and 'boolean' (i.e., true or false). Pascal/M additionally predefines the type 'string' which can contain a string of Ascii characters. A Pascal constant declaration might look like this:

```
CONST
        Data_size = 15;     (* integer *)
        Fudge     = 0.2;    (* real *)
        P_flag    = TRUE;   (* boolean *)
        Title = 'First draft'; (* Pascal/M *)
```

The next section is the type declaration. Arrays, array indices, and sets can be defined in this part. For example:

```
TYPE
  index = 1..20;
  ary   = array [indexed] of real;
  Days  = (mon,tue,wed,thu,fri,sat,sun);
```

The constant called 'index' can be used to index the elements of an array. It is defined as a restricted integer with values only in the range 1 to 20. Ellipses generally define the range of a set. ARY is declared as a real array of 20 elements. 'Days' is a set containing the days of the week.

The last part of the declaration section contains the variables. As an example:

```
VAR
  x, y, y_calc   : ary;
  i, j, k        : integer;
  a, b,
  correl_coef    : real;
  yesno          : boolean;
  Weekday        : mon..fri;
```

declares x, y, and y_calc to be one-dimensional arrays of length 20. I, J and K are integers; A, B, and correl_coef are real variables; and 'yesno' can have values of true or false. Weekday is defined as a day Mon through Fri. It does not have the values of Sat or Sun. All variables must be declared at this point.

Procedures

Subroutines in Pascal are called Procedures. Any procedures or functions that are needed by the main program must be defined at this point.

Finally, the main program can begin. This part is bracketed by the words

```
BEGIN
  . . .
    <main program>
  . . .
END.
```

A final decimal point terminates the Pascal source program. The main program might contain regular statements, or it might simply consist of procedure calls. Procedures are called by giving the procedure name and any statements. For example,

```
BEGIN
  get_data(x, y, n);
  sort_data(x, y, n);
  fit_data(x, y, y_calc, n);
  plot (x, y, y_calc, n)
END.
```

The four lines between 'begin' and 'end' are calls to procedures.

In Basic, all variables are global. This means that all values are available anywhere in the program. But a Basic subroutine that is written to sort the array X, cannot sort the array Y unless Y is first copied into X.

Variables in Fortran subroutines are local unless placed in a common declaration. Local variables are not available outside the particular subroutine. A Fortran subroutine written to sort the array X, can sort any array. The name X appears as a parameter and so is a dummy variable in the subroutine. For example,

```
CALL     SORT(Y,M)
. . . .
. . .
SUBROUTINE SORT(X, N)
. . .
```

Variables in Pascal procedures can be either local or global. All names that are declared in the main program can be global to any procedure called by the main program. For example in the program

```
PROGRAM main( output);
. . .
VAR
  n : integer;
  x : array [1..10] of real;
. . .
PROCEDURE sort;
. . .
BEGIN (* procedure sort *)
. . .
END;   (* procedure sort *)
. . .
BEGIN (* main program *)
. . .
  sort;
. . .
END.
```

procedure sort has no formal parameters and so all names in the main program are global to the procedure.

Parameters given in the procedure heading become dummy variables. They are replaced by the formal parameters of the calling statement. Furthermore, a variable declared at the beginning of the procedure becomes local to the procedure even though a variable of the same name has been declared in the calling program. The Pascal procedure

```
PROCEDURE sort (VAR x : ary;
                    n : integer);
VAR
  i : integer;
  . . .
BEGIN
<sort the array x>
END;
```

that is written to sort the array X can be called to sort the array Y of length M:

```
sort (y, m);
```

since X and N are dummy variables in the procedure.

Notice that the integer I in procedure 'sort' is a local variable. A separate variable I in the calling program will not be affected by this local variable I in procedure sort. The procedure is allowed to change the array associated with X since it is declared as a variable. The value of N, however, cannot be changed by procedure sort.

The structure of a procedure is similar to that of the main program. The declaration portion defines constants, types and variables in that order. A procedure can then declare another procedure that it will call. Each procedure automatically has access to any value in the calling program. Such a value must not be given in the procedure declaration though.

Blocks

The main part of the procedure, like the body of the main program, is bracketed with a 'begin/end' pair:

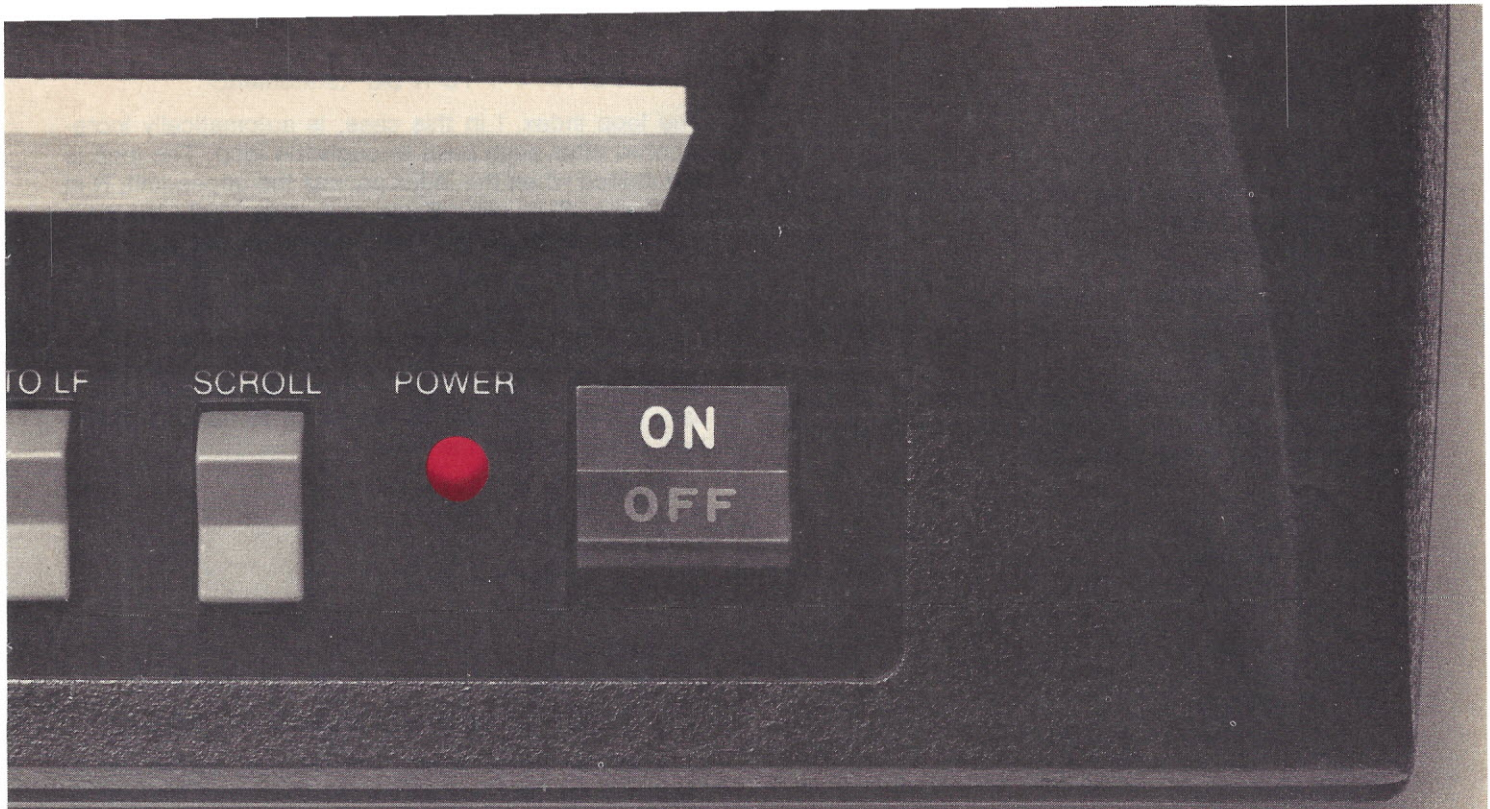
```
BEGIN
  <body of procedure>
END;
```

Within this block may be single statements like:

```
a    := 5.4;
x[3] := 4.9;
```

There may also be nested blocks such as:

```
BEGIN
  . . .
  BEGIN
```



Diablo printers spend more time in this position.

A printer isn't much good if it can't do the job when it's needed.

That's why, at Diablo, we don't just design printers that work. We design printers that keep on working. In fact, we make them so reliable, you can just open the carton, plug in and play.

Diablo offers the widest range of reliable printers and options to give the flexibility you need. Which stands to reason. After all, we pioneered the daisy wheel technology and we're still the leader in it.

So if your printers spend too much time in the "off" position, you know what to do.

Switch.

Diablo Systems

XEROX

```
    . . .
  END;
  . . .
END;
```

These interior blocks may be needed for conditional statements and loops.

The expression:

```
IF <boolean expression>
  THEN S1
  ELSE S2;
```

requires the statements S1 and S2 to be either single expressions or blocks. If multiple statements are to be executed, they will look like this:

```
IF I < 10
  THEN
    BEGIN
      FLAG := TRUE
      A    := A + 1
    END
  ELSE
    BEGIN
      FLAG := FALSE
      A    := A - 1
END;
```

Loops

There are three types of loops in Pascal. One is like the Basic 'for' loop and the Fortran 'do' loop. In fact both 'for' and 'do' appear in the heading:

```
FOR I := 1 TO N DO <statement>
```

The loop index, I in this case, is automatically incremented after each pass through the loop. The loop is terminated when the index passes the upper limit, N in this case. But unlike Basic or Fortran, only the next statement or block is repeated. The expression:

```
FOR I := 1 TO N DO
  X[I] := I;
  Y[I] := A + B * I;
```

will only act on the array X. A block structure is necessary if more than one statement is to be part of the loop. For example, the lines:

```
FOR I := 1 TO N DO
  BEGIN
    X[I] := I;
    Y[I] := A + B * I
  END;
```

will do the job.

A second type of loop uses the construction:

```
WHILE <boolean expression true> DO S1;
```

As with the 'for/do' loop, only the next statement or block is executed. Looping continues indefinitely until the boolean expression is found to be false. A block structure must be used if there is more than one statement. For example:

```
WHILE DEL > 0.01 DO
  BEGIN
```

```
    . . .
    DEL := DEL - DX
  END;
```

This loop construction is similar to the 'while/wend' construction found in CBasic and Microsoft Basic.

The third type of loop is:

```
REPEAT
  . . .
  . . .
UNTIL <boolean expression becomes true>;
```

In this case, the boolean expression is checked at the bottom of the loop. Looping continues as long as the expression is false. A 'begin/end' block is not necessary for this loop; all statements between the 'repeat' and 'until' are automatically included in the loop.

The case statement

A multibranching construction similar to the Basic 'on. . .goto' is available in Pascal. It is called the 'case' statement. Suppose, for example, that a different statement or block of statements is to be executed depending on the alphabetic character contained in the variable 'value.' Furthermore, upper-case and lower-case letters are to be treated equally. The Pascal 'case' statement might look like this:

```
CASE value OF
  'A', 'a' : R := R + 1;
  'B', 'b' : BEGIN
               X := 4.3;
               FLAG := true
             END;
  'F', 'f' : R := R - 1;
```

What if the character of 'value' is not given in the case statement? The result is undefined in standard Pascal. Some versions of Pascal allow a catch-all at the end. Unfortunately, this has not been standardized. Sometimes the default is one of the following:

```
ELSE      : . . .
OTHERS : . .
OTHERWISE : . . .
```

Input and output

Data is obtained from the console with one of two statements:

```
READ (A, B);      or
READLN (guess);
```

'Read' inputs the individual values into the corresponding variables. 'Readln' additionally reads the carriage return at the end of the line. Results are written to the console with the statements:

```
WRITE (A, B);      or
WRITELN(X, Y, YCALC);
```

The 'writeln' command includes a carriage return and a line feed at the end of the line. If the output is printed without a format specification, it will be left-adjusted like unformatted Basic. However, the output can be easily formatted by placing a colon and a field width after the identifier. For example:

```
WRITELN ('Data sets = ', N__data:5);
```

will print the value of N_data right-adjusted in a field five columns wide. Two colon/number pairs are used for real numbers. The first number gives the field width; the second number gives the number of decimal digits past the decimal point:

```
WRITELN ('% error is ', ERROR:10:5);
```

Pascal can be easy to read

One of the powerful features of Pascal is that programs can be easily understood. It is difficult to revise Basic or Fortran programs because it is difficult to understand how the program works. With Pascal, however, it is possible to write programs more clearly.

Consider the Pascal program shown in listing 1 which describes a guessing game. The computer selects random numbers in the range 1 to 100. The player tries to guess the number. After each guess, the computer tells the player whether his guess is too high, too low, or just right. If the player guesses the correct value, the computer responds with the number of guesses the player needed to find the answer. Names like 'lowest_number' and 'number_of_guesses' are self explanatory.

Pascal/M features

The predefined data type 'string' was mentioned previously. With other implementations of Pascal, the statement:

```
TYPE
  string = array[1..80] OF char;
```

must be given before defining string variables. In either case, the variable declaration might look like:

```
VAR
  Line : string;
```

String variables can be defined as in Basic:

```
Line := 'This is the data for Line';
```

Pascal/M contains many additional features that are not part of standard Pascal. The string-manipulation functions bring to Pascal all of the text-handling power of Basic. These allow string concatenation, stripping, etc. Another extra function is used to calculate the value of 10 to an exponential power. (There is no general exponent operator such as X^1.2 in Basic.) A random number generator is also provided.

Direct console control such as clear screen, home cursor, move cursor up on line, go to position x, y, etc., are possible by using a special set of functions. They are initially set for the popular ADM-3 and Soroc terminals, but can be reconfigured for other terminals.

Pascal/M is available in several CP/M versions. Some will run on both the 8080 and Z-80. Others are written in Z-80 code and so will not run on an 8080. The Z-80 version is smaller and runs somewhat faster than the 8080 version. One of the disadvantages of Pascal/M is that it is so large. A 56K byte CP/M system is required.

Pascal has been implemented on microcomputers in several different ways. For example, Pascal/MT, available from a different supplier, compiles the source program into binary code. This produces a small object program that can be run by itself. Pascal/M, however, compiles into an intermediate file that contains Ascii P-code. A run-time monitor is then used to interpret this program at execution time.

Pascal/M allows compiler directives to be placed into the source program. These directives look like comments since they are embedded in braces or parenthesis/asterisk pairs. One of the compiler directives provides an 'include' option. Frequently used procedures can be kept in separate disk files. Then the compiler can be directed to read this separate source file.

Suppose, for example, that there is a procedure in a disk file called 'sort.pas'. This will be an Ascii source file. Then the line:

```
(*$F SORT.PAS *)
```

placed in the procedure section of the main program will direct the compiler to get the procedure 'sort' from the disk file. The result will be the same as if the procedure were actually placed into the main source program.

Pascal/M is the easiest version of Pascal to use. Error messages are particularly helpful; debugging time is considerably less than with the others. □

PROGRAM LISTING

A simple Pascal program

```
PROGRAM game (input, output);
(* simple guessing game to demonstrate Pascal *)
(* computer picks a number, user tries to guess *)
CONST
  highest_number = 100;
  lowest_number  = 1;
TYPE
  number = lowest_number..highest_number;
VAR
  number_of_guesses          : integer;
  user_guess, computer_guess : number;
  answer                     : char;

PROCEDURE increment(var i : integer);
  (* just increment the integer *)
BEGIN  (* the procedure *)
  i := i + 1
END; (* procedure to increment *)

BEGIN  (* the main program *)
  REPEAT          (* the game *)
    computer_guess := trunc(100.0 * random(0));
    number_of_guesses := 1;
    write
      ('Guess a number from ', lowest_number:2,
        ' to ',  highest_number:4, ': ');
    readln(user_guess);
    WHILE user_guess <> computer_guess DO
      BEGIN
        increment(number_of_guesses);
        IF user_guess > computer_guess
        THEN
          write('Too high')
        ELSE
          write('Too low');
        write(', try again: ');
        readln(user_guess)
      END;        (* while *)
    writeln(computer_guess, ' is correct');
    writeln('Number of guesses was ',
              number_of_guesses);
    write('Do you want to try again? ');
    readln(answer);
  UNTIL (answer = 'N') OR (answer = 'n')
END.
```

Do's & Dont's in Software Selection

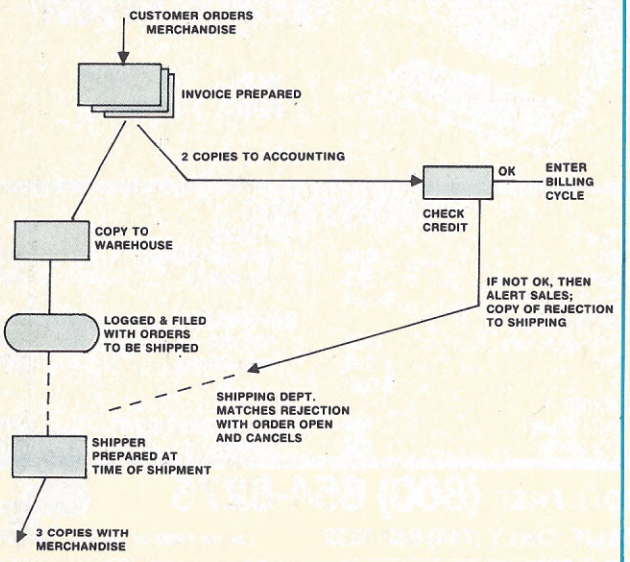
by Carl Heintz

Evaluating business software is part art and part science. The results can be critical to the success of any microcomputer installation. While many microcomputer enthusiasts are well informed about the technical capabilities of their machines, there is rampant ignorance concerning business software. Here are some ideas concerning the problem of selecting software and evaluating it.

If the shoe fits. . .

The first consideration is: critically evaluate the needs for the application. For example, putting a general ledger on a micro makes sense when there are numerous transactions. In cases where the entries are few and far between (as a trust or a small non-profit institution), the bother may not be worth it.

Many potential users view the computer as a means of eliminating the tedium of entry into a set of books. Of course, this notion is partially fallacious, since the computer will have to be fed the information. Some time may be saved through the entry process, but that will depend upon the program and the type of information.



Example of flowcharting applications.

The first step, then, is to gather relevant data about the application. Begin with a flowchart of the process being computerized (see figure). This knowledge will give the user a clear idea of what is happening in the system at the current time—whether it be service bureau, time-sharing, or just plain old manual pencil pushing.

Remember that the system is a *micro*—not a *maxi* system. Many good candidates for computerization on a micro are ruined from the beginning because of the perceived need for "icing on the cake." Get what you need, not necessarily what you want. To cite an example, users want to update inventory records, perform sales and accounts receivable updates and prepare the invoice, collection letters, credit reports, general ledger entry and sales commission reports all at one time from a little Z-80 with 32K of RAM.

What the salesman 'forgot'

Before a user can evaluate an application, there are a few facts of micro-life that the computer salesman probably didn't discuss. Among the things to keep in mind:

1. The computer is limited, in most cases, to 64K RAM. While that is obvious, many users forget that even if the whole 64K is available, there are other things that must be resident in the machine. These include: the monitor program (2K or so), the operating system (up to 30K), and the language (i.e. Basic at 10-15K). Though admittedly an extreme case, this can add up to 45K or so, leaving only 20K or so for application program and data.
2. Disk space is limited. Unless you opt for hard disk (such as a Winchester), you will be faced with the problems caused by limited storage space. The largest capacity, 5¼ disks store a maximum of 315K per disk. In most computers, the average is about 200K per disk (double density). At least two disks will be required for most applications. If you're planning on using a smaller capacity system, use four disks.
3. Be aware of the idiosyncracies of micro firmware. Although a software system may claim to run on any CP/M-equipped machine, that doesn't mean it will run the same on each system. For instance, a general ledger package offered commercially

uses the 'escape' key as a program input. On one popular microcomputer, that key automatically clears the machine and returns control to the monitor. That's not exactly what the programmers had in mind. Get it in writing that a particular program will run on a particular machine.

4. Be aware of the features of the operating system and the language you're using. For instance, in most micro versions of Basic the indexed sequential access method (ISAM) is not supported. As a result, when data is being entered, it must first be sorted before being posted to the records. In the ISAM system, the data may be posted directly to the files without sorting, since the system accesses each file and keeps an index that points to the files. While this is a simplified example, the concept is of some importance since the user may find sorting to be unacceptable.

Is home-made better?

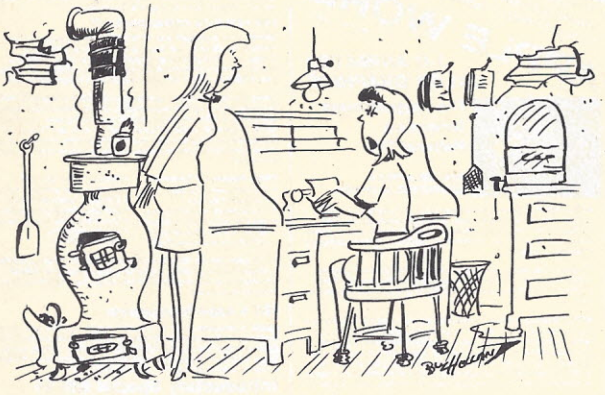
The novice computer user should ask whether it is better to build his own software program or to buy one that is pre-written, debugged (hopefully), and running. Considerations that should enter into this include:

1. Is there a commercially available software package to do the job?
2. If I make one, can I afford the time to program it, or to pay for the programming? A programmer will cost at least $10-$15 per hour and a "home brew" job may take months to program and debug.
3. If I custom program, what will be the level of error-checking compared to the sophistication and versatility of the system? For an extreme example, compare writing a word processing program to buying a program such as Electric Pencil by Michael Shrayer. He has seen far more word processing applications than the average would-be author, and he's included error trapping, versatility and sophistication that our would-be author wouldn't catch in 100 tries.
4. Consider the day-to-day user. In a small business, it may be a secretary or bookkeeper. Will she understand "home brew" documentation?
5. Is the application unique and special? Is it one in which the challenge of making it is rewarded with increased job output?

Spruce up your bug-catching

The proper detection, capture and correction of errors are critical in any software system. The majority of errors will be operator induced. The programs should have convenient effective means of dealing with them. Some features would include:

1. 'Change' routines to correct input data before it's posted.
2. Edit sequences to test for the content of certain fields (to insure, for example, that dollar fields are numeric only).
3. 'Abort' options to allow an operator to cancel a transaction entry while it's being made.
4. Reasonableness tests (such as the FICA amount on a payroll check).
5. Accumulators for debits and credits and an error warning if the amounts don't balance.



"What makes you think the boss'll never buy a computer?"

Software error detection is more difficult to delineate, but the objective of the routines should be to keep from "bombing" in the middle of a run. When operator errors result in strange data being entered, there's a chance that the software will crash. Thus, error detection from user goofs are a large proportion of software protection. Other aspects include protection from hardware errors, particularly disks, and glitches in other parts of the system. A good example of the latter occurs where the program must access a file to be sorted. Let's suppose the file exists, but is not sorted. The program should detect this and signal an error. Instead of bombing, however, we want to have it cease execution at a point that will allow an operator to intervene and hopefully straighten them out.

If the manual has no error messages or is skimpy in discussing them, frustration is inevitable. . .

The software purchaser rarely has a way to adequately evaluate software checks and error trapping without using the system awhile. Professionals who evaluate software use test decks fiendishly full of clever errors to see how the system stands up. Assuming all you have is a manual, the next best thing is to review the programmer's explanations of what the messages mean and what to do about them. In the CCA data management system, for example, each message is numbered. Beside each number in the manual is an explanation of what the error is, what it means and what to do about it. If the manual has no error messages or is skimpy in discussing them, frustration is inevitable when an error appears.

Where the responsibility lies

Implementation of a program represents the acid test to the programmer's skill. The documentation may provide some guidance, but the absence of much explanation does not necessarily mean the program is trouble. The skills of the user are often the determinant factor. If you're a knowledgeable user of CP/M, for example, Selector CIII is a snap. If you have trouble understanding how to put the diskette in the drive, you will probably get lost, no matter how well Micro-Ap documented it.

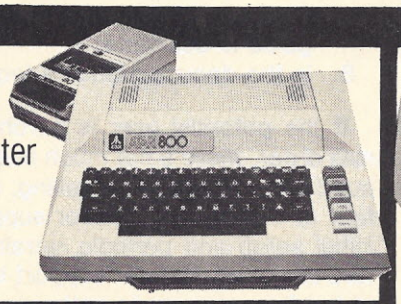
No experience can be more frustrating than being unable to locate an error code in a manual, or being unable to determine how to conclude a step in a transaction. Examining the documentation provided, the potential user should ask himself:

1. If the programmer skipped town and the software company vanished, would the user be able to solve the error?
2. Is the source code provided? If only "object" or a compiled program is sold, it's almost impossible to make changes in it.
3. Does the documentation give assistance in implementing the program?

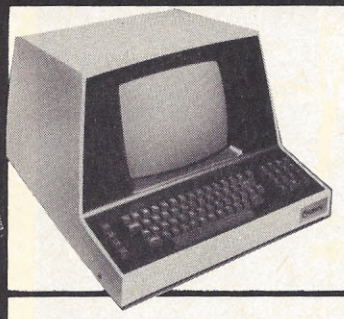
NO FRILLS! NO GIMMICKS! JUST GREAT

DISCOUNTS

MAIL ORDER ONLY

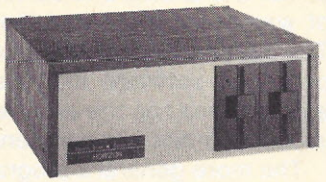


ATARI 800
Personal Computer System
$799.00



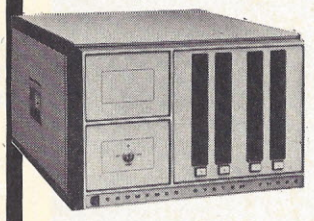
SOROC Technology

IQ 120	$699.00
IQ 140	999.00



NORTHSTAR

Horizon II 32K	$2349.00
Horizon II Quad	2799.00
Horizon II 64K	2999.00
Horizon Quad 64K	3399.00



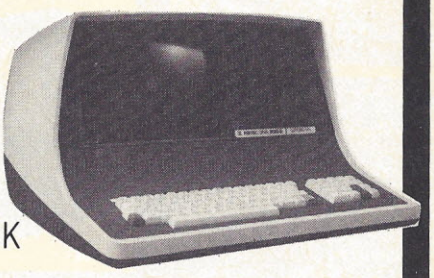
CROMEMCO

System 3	$5695.00
Z2H	7995.00

TELEVIDEO

912 **$749.00** | 920 **$799.00**

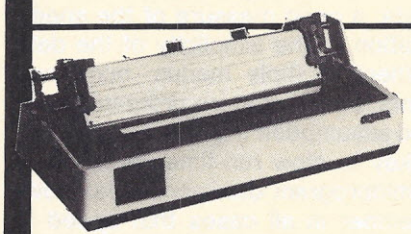
INTERTEC
Superbrain 32K Computer
$2495.00
Superbrain 64K
$2795.00



HAZELTINE

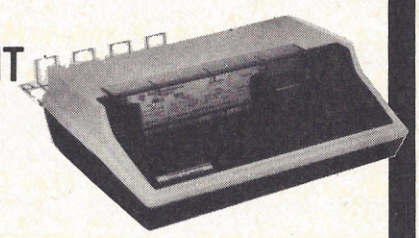
1420	$795.00
1500	$849.00
1510	$1049.00
1520	$1229.00

DECwriter IV
LA 34 **$979.00**



OKIDATA
Microline 80
$699.00

TEXAS INSTRUMENT
810 Multi Copy Impact Printer
$1499.00



We'll meet or beat any advertised prices!

Most items in stock for immediate delivery. Factory-sealed cartons.
Full manufacturer's guarantee.

DATA DISCOUNT CENTER

Box 100 135-53 Northern Blvd., Flushing, New York 11354

Visa • Master Charge • N.Y.S. residents add appropriate Sales Tax • Shipping F.O.B. N.Y.

PHONE ORDERS CALL 212 465-6609

4. Are sample error messages shown and does the manual adequately discuss what they mean?
5. Is a description of the program and its major features included? Is a flowchart included?
6. Are file types, arrangement, and contents discussed?
7. Is an operator's procedure guide given? Even in the case of fully menu-driven programs, such a guide is desirable.
8. Is the documentation adequately indexed?

If the potential user is purchasing the software by mail order, there's not much use in discussing training; but if a local supplier is selling, a key inquiry is the extent of training and vendor support. Training includes initial setup and periodic retraining when new operators are hired. Vendor support includes the availability of hot line answers to software questions, the degree of warranties given and the assistance in installation. The purchaser should ascertain in the case of prepackaged software that the vendor understands the software and has the experience necessary to assist in implementation and maintenance.

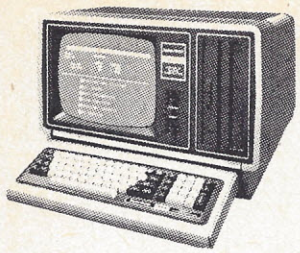
The more general a program, the less likely it is to fit anyone's specific needs. On the other hand, some degree of flexibility is important. As a user's needs change, the program must have the capacity for enhancement. In the case of a general ledger program, for example, the user may wish to add branches or add accounts. The programs represent substantial investments which will be rendered useless unless they have this capacity.

On the other hand, a potential user should not overlook the possibility of purchasing an adequate system for present needs with the full realization that in three or four years a whole new software system will be purchased. Considering the advances in the art to date, better products will probably be available.

Fastest isn't always best

Hardware enthusiasts speak in terms of clock frequencies, disk access times and printer throughput; but for software, there are no objective standards by which one application program can be adequately compared against another. However, a feel for the ability of the system can be gained by an examination of the input procedures, the degree of "chaining" within the programs, and the logical flow of information from manual to computer to manual systems. Realizing that most time will be spent inputting data, a good measure of the speed can come by examination of the simplicity of the data entry sequences. Where multiple menus, numerous repetitive data elements, or two or more programs must be utilized, it is a good assumption that the program will run slowly. Another key to slow run-times is the presence of numerous midprogram disk accesses. However, one must remember in all cases that speed is comparative and the first essential is comparability.

Evaluating business software is an important part of microcomputer selection and installation. The success of the installation will be enhanced by informed software choices made by users who utilize a number of diverse criteria to select the right software. By specifically defining needs and critically evaluating information, the user can go a long way toward insuring a successful choice. □



AUTHORIZED TRS-80® DEALER A301

COMPUTER SPECIALISTS

26-1056 16K Level II System with Keypad.	$ 700.00
26-1145 RS-232 Board	84.00
26-1140 "O" K Interface	254.00
26-1141 "16" K Interface	381.00
26-1142 "32" K Interface	508.00
26-1160 Mini Disk - Drive 0	424.00
26-1161 Mini Disk - Additional	424.00
26-1154 Lineprinter II	720.00
26-1156 Lineprinter III	1799.00
26-1157 WP50 Daisy Printer	2670.00
26-1180 Voice Synthesiser	339.00
26-1181 VOXBOX	145.00
26-1104 Factory Upper/Lower Case Modifaction Installed	90.00
26-1605 Scripsit - Tape	60.00
26-1563 Scripsit - Disk	85.00

10% DISCOUNT Off List
26-4002
64K 1 Drive
$3499.00

15% DISCOUNT Off List
26-1054
4K Level II
$527.00

ALL OTHER R.S. SOFTWARE FURNITURE, STANDS, CABLES AND ACCESSORIES DEDUCT 10% FROM CATALOG PRICE

LOBO drives INTERNATIONAL

Plug Compatible Lobo 5¼ Drives - $375.00

Versatile Lobo Interface, 8" Drives and Hard Drive.
Call For Prices

CENTRONICS

Fast 100 CPS Centronics 730 Printer - $675.00
Text Quality Centronics 737 Printer - $850.00

Novation Cat Modem . . $159.00
CCA Data Management System 72.00
Adventure Games Games 1-9 . . . . . 14.00

MICROSOFT

Model I Basic Compiler	$195.00
Model II Basic Compiler	395.00

BASF

10-5¼	Diskettes	$45.00
10-8	Diskettes	47.00

Acorn Software Products, Inc.

GAMES:	
Alien Invasion	$9.00
UTILITIES:	
System Savers	14.00
EDUCATION:	
Language Teacher	18.00

1-800-841-0860 Toll Free Order Entry

MICRO MANAGEMENT SYSTEMS, INC.

No Taxes on Out Of State Shipments

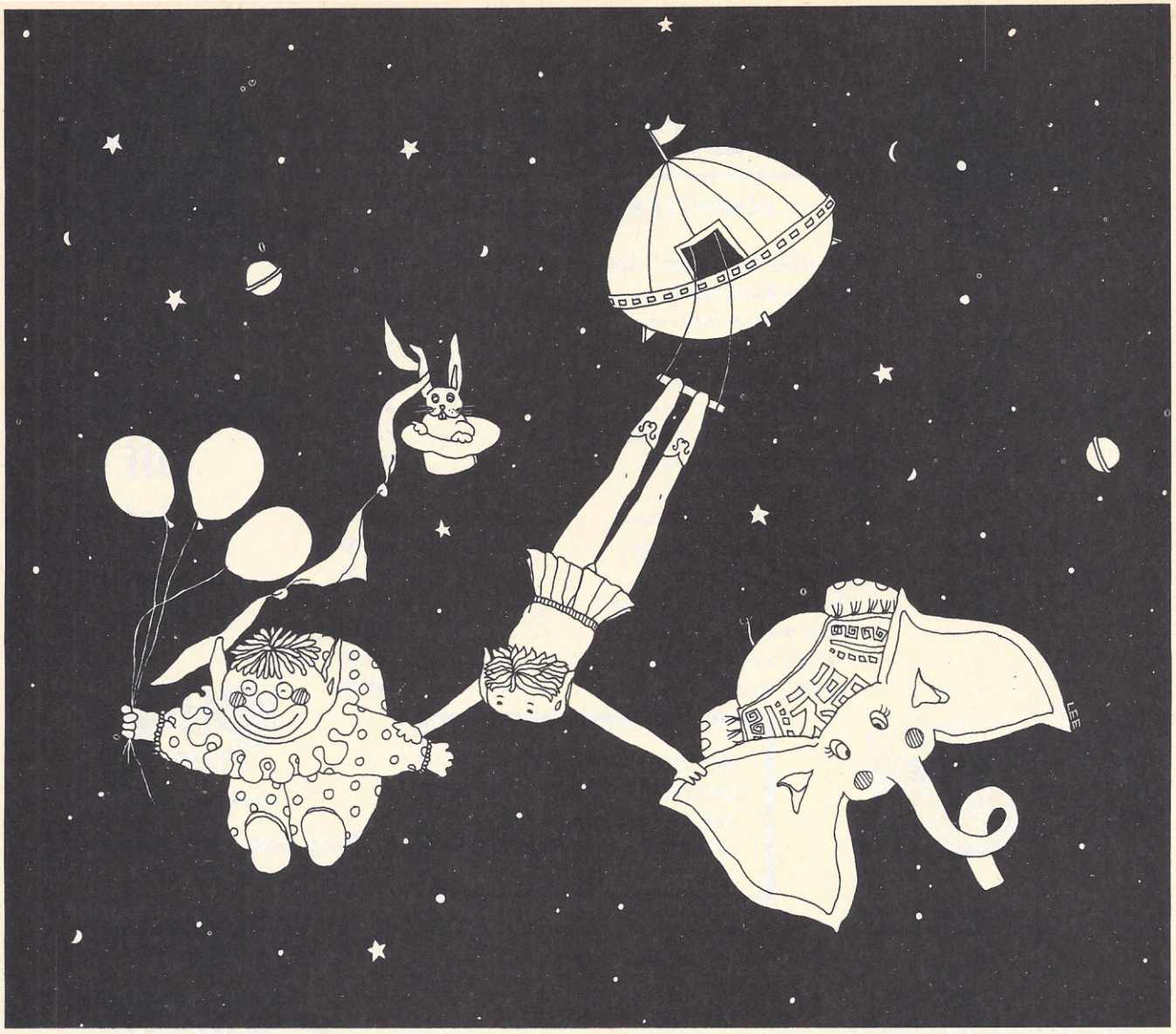
Immediate Shipment From Stock.

DOWNTOWN PLAZA SHOPPING CENTER
115 C SECOND AVE. S.W.
CAIRO, GEORGIA 31728
(912) 377-7120 Ga. Phone No.

Full Factory Warranty on All Items Sold.

Largest Inventory In the S.E. U.S.A.

*TRS-80 is a registered trademark of the Tandy Corp.



Headlining Interstellar News

by Ralph Roberts

Here's a program that lets you subscribe to the Interstellar News Service. That's right, the news of the future. . .today. Why wait?

This powerful little program will give you monkey barrels full of fun. And, if you write science fiction stories as I do, here's a wealth of ideas that'll come tumbling out of your computer like an errant storm of wobbly space debris.

The program, itself, is pretty simple and is written in Smoke Signal Basic for my Chieftain computer. However, it should run on most systems with very minor changes. You may want to add to the number of nouns, verbs and adjective/adverbs. Just add more data lines to do so and be sure to change the constants in line 0010 (N is nouns, etc.).

The headline generated follows the pattern noun, verb, adjective or adverb, noun. As the program stands, it should generate over ten million possible combinations of readable and sensible headlines (56 x 56 x 59 x 56 = 10361344).

Plug this program into your computer and you can tell your friends things like: "planet Mars gives birth to fantastic little green man. . ." or "traveling space circus blasts out of creeping delapidated starship. . ."

AP and UPI. . .eat your hearts out. □

PROGRAM LISTING

```
*
0001 REM ::: FUTURE NEWSPAPER HEADLINES PROGRAM :::
0002 REM ::: by Ralph Roberts ::::::::::::::::::::
0010 LET N=56:LET V=56:LET A=59
0012 DIM N$(N),V$(V),A$(A)
0020 FOR X=1 TO N:READ N$(X):NEXT X
0022 FOR X=1 TO V:READ V$(X):NEXT X
0025 FOR X=1 TO A:READ A$(X):NEXT X
0050 INPUT "How many future newspaper headlines do you want",Q
0055 PRINT :PRINT
0060 PRINT "From the computers of Interstellar News Services,"
0065 PRINT "          Here are today's news stories .... "
0070 PRINT :P.
0100 FOR X=1 TO Q
0110 GOSUB 500
0120 PRINT N$(N1);" ";V$(V1);" ";A$(A1);" ";
0130 GOSUB 500
0140 PRINT N$(N1)
0145 PRINT
0150 NEXT X
0155 INPUT "Enter a zero (0) to end program, a one (1) to for more",I:IF I=0 END
0199 GOTO 50
0500 LET N1=INT(RND*N)+1:V1=INT(RND*V)+1:A1=INT(RND*A)+1:RETURN
0990 REM NOUNS
1000 DATA SHIPLOAD OF VEGAN COLONISTS,BIONICALLY MODIFIED HUMAN,BLACK HOLE
1005 DATA GIGANTIC ALIEN BEAST,FOUL SMELLING REACTOR FUEL,DAMAGED SPACECRAFT
1010 DATA BLASTER TOTING ROBOT,RETIRED LADY ASTRONOMER,MICROPROCESSOR BASED ANDROID
1015 DATA WARPSHIP CAPTAIN,DERANGED SPACESAILOR,MULTI-WORLD CORPORATION,MONSTER
1020 DATA PLANET MARS,MOONS OF JUPITER,EARTH'S MOON,MILKY WAY GALAXY
1030 DATA HUGE COMPUTER,LOWLY SPACEBUM,EMPEROR OF RIGEL IV,SPACESHIP EARTH
1040 DATA MYSTERIOUS ALIEN BEING,CREW OF UFO,WARLORD OF EIGHT PLANETS,LITTLE GREEN MAN
1050 DATA SPACE FREIGHTER NAVIGATOR,BONDED ROBOT LAWYER,AMOROUS ANDROID,COMET
1060 DATA CLOUD OF SPACE DEBRIS,SWARM OF METEORS,TYPE G SUN,INTERSTELLAR BEAUTY QUEEN
1070 DATA DELAPIDATED STARSHIP,BIG PILE OF VEGAN SWAMPWEED,RINGS OF SATURN,STAR
1075 DATA DEFEATED SPACE HOCKEY CHAMPION,FEDERATION BATTLECRUSIER,CRAZY BEAST
1077 DATA ONE-ARMED ROBOT,CRAZED WARPLINER CREWPERSON,MARAUDING SPACE PIRATES
1080 DATA USED STARSHIP SALESMAN,TRAVELING SPACE CIRCUS,SPACEWHALE,MOON MERMAID
1085 DATA INTERSTELLAR PLAYBOY,WELL KNOWN GARVITZ IMPORTER,SPACE PEBBLE,ROCK
1090 DATA STAR SCIMMER MANUFACTURER,POPULAR STAR-ROCK BAND,ALIEN CRIMINAL,NURD
1091 DATA FAMED INVENTOR
*
2000 REM VERBS
2001 DATA BURNS,OVERRUNS,LEAPS,FIGHTS,LOVES,ESCAPES,EATS,SLAMS INTO,BLASTS OUT OF
2005 DATA BUYS CONTROL OF,GIVES BIRTH TO, MARRIES,CARRIES OFF,CHEATS,DEFEATS
2008 DATA FALLS IN LOVE WITH,LIVES WITH,DISASSEMBLES,SHOOTS,DIGESTS,INSULTS
2010 DATA KISSES,RIPS THE COVERING FROM,PEEKS AT
2015 DATA ADDRESSES,RENTS,SELLS,BUILDS WORKING MODEL OF,PHOTOGRAPHS,LIVES IN
2019 DATA EATS PICTURE OF,DEFACES PICTURE OF,MAKES WHOOPEE WITH,INTIMIDATES
2020 DATA DONATES GIFTS TO,SENDS FLOWERS TO,HAS DATE WITH,CAUGHT NECKING WITH
2030 DATA PAINTS PICTURE OF,SLEEPS IN,ACQUIRES,LOSES,MAKES OUT WITH,DIVES INTO
2035 DATA CONSTRUCTS INFLATABLE MODEL OF,COLLIDES WITH,WEATHERS SPACE STORM WITH
2040 DATA FALLS INTO SUN WITH,DANCES A JIG WITH,CURSES,DOES IMITATION OF,LICKS
2050 DATA PICKS UP IN BAR,TAKES ADVANTAGE OF,MAKES ADVANCES TO,BARFS ALL OVER
*
3000 REM ADVERBS AND ADJECTIVES
3001 DATA DASTARDLY,PRETTY,LOVELY,FIERCE,RAMPAGING,FANTASTIC,UNBELIEVABLELY HUGE
3005 DATA SMOKE TRAILING,VOLUPTUOUS,BARBARIC,WEALTHLY,IDIOTIC,NEWLY CREATED
3007 DATA LONESOME,DELECTABLE,UGLY AND MOROSE,FAT,UNREPENTING,MICROSCOPIC
3010 DATA FLAMING,RADIOACTIVE,SATANIC,ANGELIC,WELL EDUCATED,SHAPELY,DELIGHTED
3020 DATA KNOWLEDGEABLE,STUPIDLY STUBBORN,WILLINGLY CO-OPERATIVE,DUMB,LIKEABLE
3030 DATA ASININE,DOPE SMOKING,WEIRD,FEISTY,FLAT-CHESTED,HOPELESSLY INSANE
3040 DATA WELL ENDOWED,QUICK THINKING,FAST TALKING,ENGLISH SPEAKING,FOUL
3050 DATA EXCEPTIONAL,SPACESHIP STEALING,BRAINY,WAR-MONGERING,PROVERB SPOUTING
3060 DATA MENACING,SLENDER,POLITICALLY MOTIVATED,LUST MINDED,UNCOUTH,BRAINLESS
3070 DATA SILVER-TONGUED,DRUNKEN,TIPSY,SLOWLY CREEPING,FREE SPIRITED,MADLY DEVIOUS
```

NEW PRODUCTS

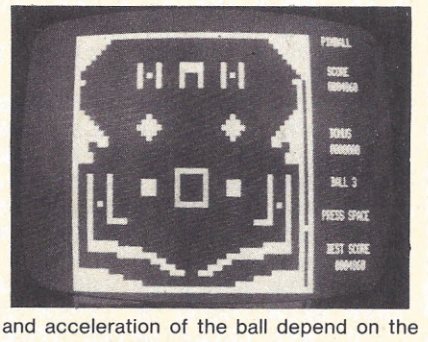
Dental office system, the CP/M version of Dentistaid, is designed to streamline all major time-consuming tasks performed in the dental office. It is completely menu-driven. Some features include print standard ADA insurance forms, pre-qualification and actual services; monthly statements including envelope and return envelope; patient recalls for mailing and telephone follow-up; accounts receivable aging reports with no payments made and balance over 90 days indicators. The program is written in CBasic2 running under CPM for 8", 5¼" and Micropolis disk drives. Minimum RAM is 32K. A standard CRT with an RS-232 connector, and a serial or parallel 132-column printer, along with 360K of storage on two disk drives, are required. Using this configuration, the system will support 1,500 accounts with each account supporting 6 individuals. Micro Computer Management Inc., P.O. Box 1794, Ft. Collins, CO 85022, (303) 493-5700.
CIRCLE INQUIRY NO. 121

Elementary math educational disk, written and designed by a professional educator, contains an arithmetic readiness test and four interactive lessons designed to teach elementary addition, subtraction, multiplication and division on nine different skill levels. Lessons offer interactive tutorials and use color graphics and computer simulated voice to maintain student interest and reinforce basic concepts. Student scores are maintained on disk and are accessible only through a special teacher program (included). The Edu-disk is self-demonstrating and requires little or no instructor assistance. Recommended for the student with no prior arithmetic experience, and as a suppliment in higher level remedial situations. Price: $39.95/disk. Muse Software, 330 N. Charles St., Baltimore, MD 21202, (301) 659-7212.
CIRCLE INQUIRY NO. 122

Bookkeeping products for CP/M and TRS-DOS operating systems, Bookkeeper I, written in Microsoft Basic, includes general ledger, accounts receivable (balance forward), accounts payable, and payroll. All are modular (stand-alone) in design but share consistent screen formats for user prompting and ease of use. The line is supported by excellent nontechnical operator reference manuals for user product training and self-maintenance. Maintenance for tax tables, W2s, quarterly tax reports, financial statement headings, etc. requires no programming and may be performed by the user. Two versions are available: One for the TRS-80 I 32K, dual disk business system, and the other for the CP/M operating system in conjunction with Microsoft Basic. Data Train Inc., 840 NW 6th St., Suite 3, Grants Pass, OR 97526, (503) 476-1467.
CIRCLE INQUIRY NO. 123

Business accounting package, Real-Tabs, written in North Star Basic for real estate, management brokers, attorneys, title companies and mortgage brokers, operates on a real-time mode making it possible to obtain up-to-date reports (printed or viewed) at the end of each business day. Payroll is fully supported along with summary reports for year-to-date and for any specified time period. A database management system is included for profitable managerial control. Ordered files provide instant retrieval of the business history and identity of any client or customer. In addition, particular attention is given to client escrow accounts and separate bookkeeping histories. Vincent D. Puzar, 5905 Gulf Blvd., St. Pete Beach, FL 33706, (813) 360-0845.
CIRCLE INQUIRY NO. 124

Arcade game, Pinball, for the Radio Shack I level II TRS-80 written in machine language includes flippers, bumpers, rollovers, runs, bonus points. The space bar on the TRS-80 releases the ball at various speeds under player control. Once in play, both the speed

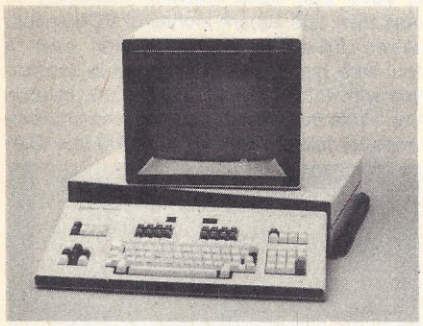


and acceleration of the ball depend on the contact with various features on the board, including the "Bermuda Square." Priced at $14.95 on cassette, or $20.95 on disk. Acorn Software Products Inc., 634 North Carolina Ave., S.E. Washington, D.C. 20003, (202) 544-4259.
CIRCLE INQUIRY NO. 125

Development software comparable to that offered by the microprocessor manufacturer enables any CP/M system to serve as a development station for the Intel 8048 series, RCA Cosmac 1802/1804, National COP400 series and the Zilog Z-8 processors. These development systems feature a macro-assembler, an interactive editor/assembler and a text editor.

The systems share a common operational structure, with uniform procedures for program entry, modification, assembly and disk file handling. The macro-assembler includes full macro and conditional assembly features as well as the ability to chain a series of source files together during a single assembly. Programs developed under these systems must be off-loaded to the target processor for test. Facilities are provided to implement the off-loading mechanism as a direct transfer from memory, via a byte stream over a CPU port, or via '.com' or '.hex' disk files. Each system is $150 on CP/M 8" soft sector (3741), 5" North Star or 5" Micropolis Mod II (Lifeboat adaptation) diskette, with complete documentation. Allen Ashley, 395 Sierra Madre Villa, Pasadena, CA 91107. (213) 793-5748.
CIRCLE INQUIRY NO. 126

Multi-function terminal, SST 300, features a 15-inch diagonal screen capable of displaying 30 rows of 132 characters. Thirty additional rows are available for display scrolling. A dual display function of the terminal allows two different character sets (Ascii and user defined) to be displayed simultaneously. An optional RAM to define



sets may be entered either by a computer or through the keyboard. Another standard feature is an 8K buffered printer port which allows the user to print an entire screen of information without interfering with normal terminal operations. Tridata Systems, 1206 John Reed Ct., Industry, CA 91745, (213) 330-1691.

CIRCLE INQUIRY NO. 149

A wireless keyboard input for TRS-80, the model RX-10, includes a hand held, ultrasonic remote which can be used as an adjunct to the standard keyboard for convenient data entry from across the room. Basic programs can read ultrasonic input easily with a few minor changes. The system also permits control of remote devices located anywhere

within the home or office. Communication to the remotes is via the AC power lines. A flexible scheduling program can activate the remotes automatically using cyclic, time of day, or future data schedules. Included are cables, interfaces, cordless controller, command console, appliance and lamp control modules. Software is provided on diskette for status display, security monitoring, and scheduling. Priced at $285. Omni Automation, P.O. Box 7716, Atlanta, GA 30357, (404) 581-0284.

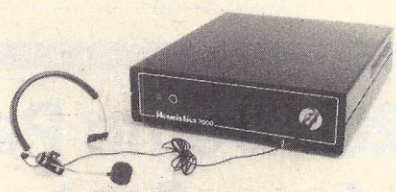
CIRCLE INQUIRY NO. 128

An acoustic coupler package for minicomputers, called the Clear Signal, is packaged with any of 34 standard cables for use with Hewlett-Packard, Data General, Texas Instruments, and Digital Equipment Corp. hardware. The coupler operates in full and half duplex modes and has an acoustical self-testing feature that eliminates the need for remote assistance in diagnosing most transmission problems. Originate/answer modes are switch selectable.

It also has a sensitivity of -50 dBm, which exceeds standard telephone voice-service specifications. It is Bell 103A compatible and interfaces with most EIA RS 232C computers and terminals at 0-300 baud. Inmac, 2465 Augustine Dr., Santa Clara, CA 95051, (408) 727-1970.

CIRCLE INQUIRY NO. 129

Speech recognition unit is making available speech input capability with virtually every computer terminal. The 7000 will enable technicians, business executives and others who don't type, or are busy with other tasks, to enter information into their computers, directly and with no errors. Key to the unit is a spectrum analyzer that uses digital filtering and pattern matching techniques to analyze audio input. The output is



automatically transferred to the computer in standard Ascii format. It can be trained to recognize up to 64 words or phrases, each up to three seconds in length, and is compatible with all common programming languages. It can be trained or re-trained as often as necessary to accept the voice or voices of users, and will automatically reject utterances significantly different from the vocabulary set. Price is $3,000. Heuristics Inc., 1285 Hammerwood Ave., Sunnyvale, CA 94086, (408) 734-8532.

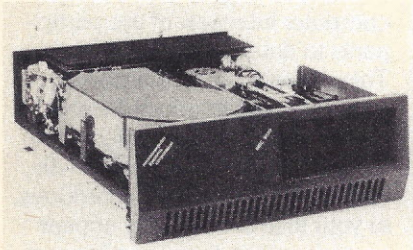
CIRCLE INQUIRY NO. 130

The Nobus-Z microcomputer features a 4 MHz Z80A CPU, CP/M operating system, 64K dynamic RAM, dual density 8-inch disk drives with 600K bytes/side, and 6K color text/graphics. The system can support both personal and business applications. Console configurations range from a keyboard and TV set to separate word processing display terminals. A full line of printers and hard disks is available. Exo Electronics Co., P.O. Box 3571, Culver City, CA 90230, (213) 390-6527.

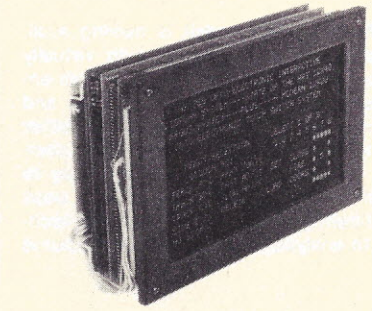
CIRCLE INQUIRY NO. 131

A 5.25-inch minidisk, the MD 577 Super Mini, is compatible with virtually every minidisk drive. The disk is certified for 40 tracks and then recertified for 77 tracks. It features a hub reinforcement ring designed to create better centering ability and to reduce the possibility of hub damage. The MD 577 has a 250 kbit/second data rate capability, a capacity of 481.3 kbytes (unformatted), a maximum recording density of 5223 fci ("double density") and track density of 100 tpi. Verbatim, 323 Soquel Way, Sunnyvale, CA 94086, (408) 245-4400.
CIRCLE INQUIRY NO. 132

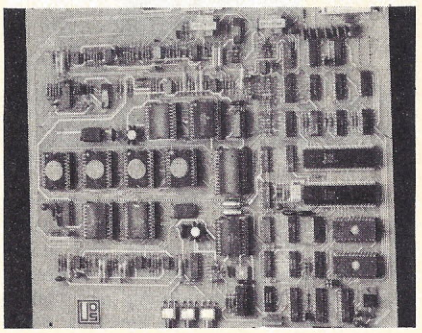
Disk storage system, MSC-8100, combines hard-disk mass storage with flexible-disk backup. The system provides an economical way of expanding the on-line storage capacity of most small computers, while retaining the media transportability of floppy disks. The unit is a self-contained data storage system that incorporates: an intelligent controller/formatter with a universal IEEE-488 bus protocol, a high-density Winchester technology drive using 8-inch, fixed disks for capacities to 19.1 megabytes, and a backup flexible-disk drive with a capacity of 1.6 megabytes per disk. The system is seven inches high and can be mounted in a standard 18-inch rack. Priced at $9,250. Microcomputer Systems Corp., 432 Lakeside Dr., Sunnyvale, CA 94086, (408) 733-4200.
CIRCLE INQUIRY NO. 133



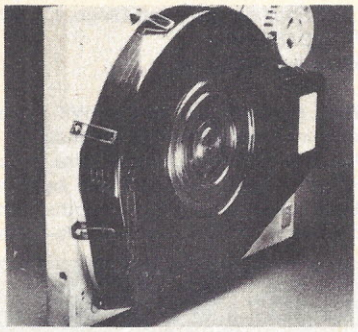
Alphanumeric display subsystem with interactive capability for direct operator interface allows data entry by touching the surface of the display at the desired location as indicated by the message displayed beneath it. The ability to erase or change the displayed message by application of logic input signals allows the device to take on any desired control function. A dot matrix message panel provides a display of characters in 5x7 format with underline and forward-cursor capability. There are two display configurations available: 256-character (8 rows of 32 characters) and 480-character (12 rows of 40 characters). Character size is .18x.26 and .15x.21 inches, respectively. Opto and transparent switch array technology provide discrete switch capability for every other character position within each row. IEE, 7740 Lemona Ave., Van Nuys, CA 91405. (213) 787-0311, ext. 206.
CIRCLE INQUIRY NO. 134



A 9600 bps OEM modem, packaged on less than 100 square inches of PCB space, allows full-duplex communication over a four-wire type 3002 circuit, and is compatible with CCITT specification V.29. In addition to its 9600 bps capability, the unit provides for fallback to 7200 or 4800 bps. Carrier frequency is 1700 Hz (±.01%) and line impedance is 600 ohms, transformer-coupled, with transient protection. Digital interface conforms to RS-232C and CCITT V.24. Control reposes totally in the software, providing easy adaptation to individual system needs. Universal Data Systems, 5000 Bradford Dr., Huntsville, AL 35805, (205) 837-8100.
CIRCLE INQUIRY NO. 135



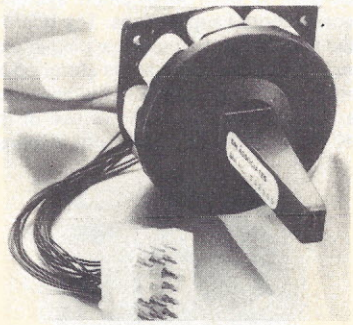
Winchester fixed disk drive at 58 megabyte, 14 inches for extensive mass storage requirements, mounts in a 19-inch rack while using only 5.25 inches of panel space. Weight is 40 pounds. Specifications include a transfer rate of 7.1 megabytes per second, average access time of 65 milliseconds and average latency of 10.1 milliseconds. Error



rates are one per 10^{10} bits for soft read errors, one per 10^{12} bits for hard read errors and one per 10^{6} bits for seek errors. Recording density on the SA4100 is 5534 bits per inch with a track density of 172 tracks per inch. Mean time between failure is 8,000 power-on hours of typical usage. The unit operates without the need for preventative maintenance and offers a component design life of five years. Shugart, 435 Oakmead Pkwy., Sunnyvale, CA 94086, (408) 733-0100.

CIRCLE INQUIRY NO. 136

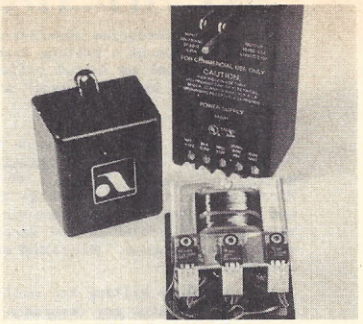
Impact matrix print heads are available for OEMs in the production of printers as well as for field service replacement on certain popular matrix printers. The Model 1000 has a maximum print wire frequency of 1,250 Hz, typical life of 300 million characters, and is designed to operate over a wide range of voltages and pulse widths. The Model 5000



is 100% electrically and mechanically compatible with Lear-Siegler impact matrix printers, and is interchangeable in the field as well as in ongoing production. The Model 6000 is 100% compatible with Diablo impact matrix printers as well as with the UMI (Universal Microprinters) print head. Prices range from $150 (quantity one) to $65 (10,000 and over). DH Assoc., 754 N. Pastoria Ave., Sunnyvale, CA 94086, (408) 738-2082.

CIRCLE INQUIRY NO. 137

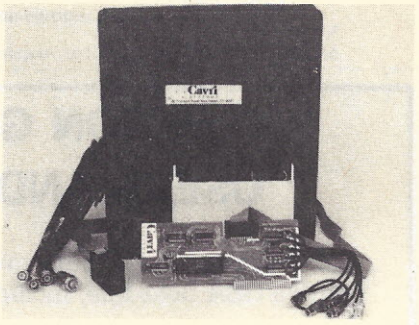
Triple-output power supplies are microprocessor-based data-communications applications—such as modems and multiplexers. The power supply is available in two models: One supplies +5 VDC at 0.30 amperes, and ±12 VDC at 0.13 amperes; the other furnishes +5 VDC at 0.60 amperes, ±12 VDC at 0.20 amperes. Such external power supplies eliminate the problems of heat buildup,



RF interference and space requirements. The power supply plugs into any 117-volt AC outlet, or is available with a line cord. Line/load regulations is ±5%, with less than 10mv RMS ripple. Overall dimensions of the small model are approximately 3 x 3 x 2 inches; the larger unit measures 5 x 3 x 2 inches. Impact-resistant cases are black, with other colors optional. Ault, Inc., 1600 H Freeway Blvd., Minneapolis, MN 55430, (612) 560-9300.

CIRCLE INQUIRY NO. 138

Computer/video player integrator, the Cavri III, enables a user, seated at the keyboard, to index and later access videotape frames or to interact with videotaped materials. The system, used for comprehensive storage and retrieval of text and audio-visual information, also allows control of all remote functions of the VCR. The



system consists of an Apple I/O board, cables and connectors, system software in Applesoft Basic on disk and a user's manual. It is available for VCRs that carry a control pulse or that interface with manufacturer's search units. Cavri Systems Inc., 26 Trumbull St., New Haven, CT 06511, (203) 562-9873.

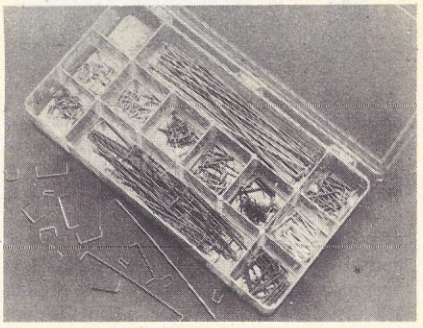
CIRCLE INQUIRY NO. 139

Front panel console for DEC LSI-11 based systems supports either real-time or static debugging of software and hardware. The RPC-11 allows users to examine or alter

memory and registers as well as perform all other standard ODT functions while a program is either running or halted. The dual-width interface card replaces a standard DLV-11 console interface. Power, supplied by the host computer, is 5 volts at 1.5 amps maximum. The RPC-11 is lightweight and portable. Priced at $1,195. Metacomp, Inc., 7290 Engineer Rd., San Diego, CA 92111, (714) 278-0635.

CIRCLE INQUIRY NO. 140

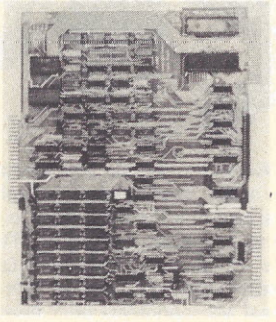
Wire jumper kit, Model WK-1, makes solderless breadboarding much easier. It includes twenty-five pieces of wire, precut to defined lengths, stripped and bent into a staple shape, in each of 14 lengths: from 0.1 to 1.0 inch in 0.1 inch increments—plus 2.0,



3.0, 4.0 and 5.0 inch lengths. Each wire is length-coded with standard color-code insulation. All wire is AWG #22 solid. The wires are packaged in a compartmented plastic box with a hinged lid. Price is $10. Global Specialties Corp., 70 Fulton Terrace, New Haven, CT 06509, (203) 624-3103.

CIRCLE INQUIRY NO. 141

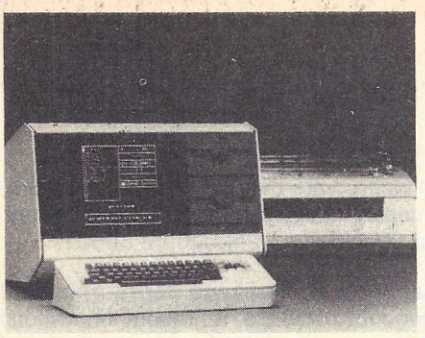
Computer board for Rockwell's Aim-65, the Memory-Mate, offers compact, reliable expansion for both development system and process control applications. The board provides RAM sockets that allow plug-in expansion in 8K or 16K increments assigned in 4K blocks, each block positionable anywhere in the system. The board allows use of 8K or 16K RAM chips—or both—for expansion.



Features include full parity check circuitry, checking memory integrity continuously, and protection for on-board RAM. Any memory cell failure results in an immediate LED indication and/or program interrupt. Forethought Products, 87070 Dukhobar Rd., Eugene, OR 97402, (503) 485-8575.

CIRCLE INQUIRY NO. 142

Desktop full word and data processor, the Alphasprint, includes 64K of main memory, a 45K display buffer, high resolution 12-inch video display, and two double density 5¼-inch diskette drives storing over 200 pages of text or 330K of data. A Selec-



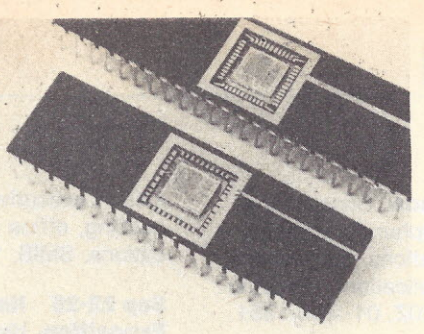
tric II-type keyboard incorporates 72 character and function keys, and a full numeric keypad. Word processing is geared toward long document preparation, offering high-speed automatic global search and replacement of words or phrases as well as block move/delete/insert. The display buffer, storing 24 pages, is larger than standard, lessening the need to page to disk. Alpha Professional Systems, 9465 Wilshire Blvd., Beverly Hills, CA 90212, (213) 377-6703.

CIRCLE INQUIRY NO. 143

Computerized video camera system acquires and stores data under control of a DEC PDP-11 system. The MIP-3/V contains a high-speed microcomputer-controlled input processor board, a 16K-word dual-port memory board and a 128/128x8-bit per pixel solid-state camera equipped with a 25mm f 1.4 lens and 25-foot cable. The system inputs up to 30 frames of data per second, enabling the user to acquire data continuously or perform real time processing such as integration. The camera buffers up to two complete frames and is expandable to eight frames of data. Price: $9,500. Computer Design and Applications, 377 Elliot St., Newton, MA 02164, (617) 964-4320.

CIRCLE INQUIRY NO. 144

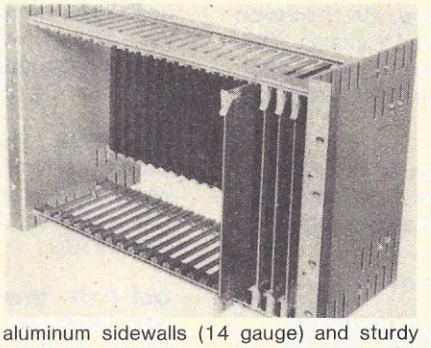
Microprocessor circuit, 16 bit, will run at a clock rate of 6.0 MHz and is available in two package types for different applications. The segmented Z8001A, in a 48-pin dual-in-line package, permits the user to address up to 8 megabytes of memory for highly



memory-intensive applications. The non-segmented 40-pin Z8002A allows addressing of 64 kilobytes of memory for less memory-intensive uses. Zilog, 10340 Bubb Rd., Cupertino, CA 95014, (408) 446-4666.

CIRCLE INQUIRY NO. 145

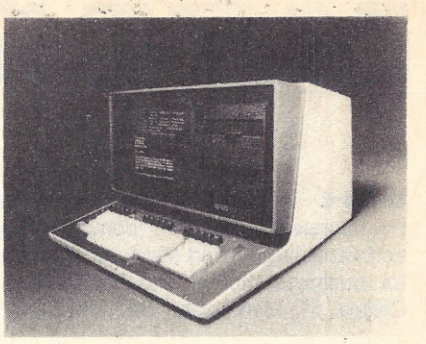
EIA standard rack-mounted cage holds 21 S-100 bus cards on ¾-inch centers for packaging microcomputer systems. The CCK100 has adjustable struts for mounting screw-down card-edge connectors or the Vector 8803 S-100 bus motherboard without hole driling or special hardware. Fabricated of 0.081-inch clear anodized



aluminum sidewalls (14 gauge) and sturdy extruded aluminum cross members, the cage measures 19 x 12.2 x 8.9 inches. Price: $49.80. Vector Electronic Co., 12460 Gladstone Ave., Sylmar, CA 91342, (213) 365-9661.

CIRCLE INQUIRY NO. 146

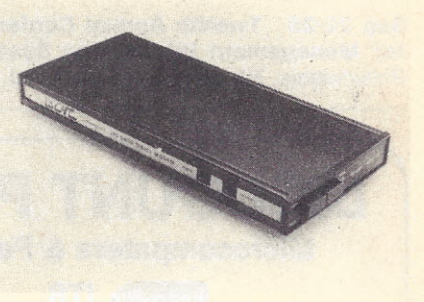
Desktop computer for OEMs and systems houses, System 10, includes a Z80 processor, 65K RAM, 700K disk storage, hard disk interface, DMA controller, interrupt



controller, and three serial I/O channels. Operator interaction is handled through a Selectric-style keyboard with a 10-key pad and downloadable function keys. The display is 80 columns by 25 lines featuring a real-time clock. A separate microprocessor handles the CRT. Software support consists of CP/M version 2 disk operating system and a screen editor. Basic, Fortran, Pascal, C, Cobol and other supported software are optionally available. Price: $4950 single quantity; $3465 in 100s. Gnat Computers, 7895 Convoy Ct., San Diego, CA 92111, (714) 560-0433.

CIRCLE INQUIRY NO. 147

Direct connect modem for home or office can function on either a multi- or single-line phone. The D-Cat is a Bell 103 compatible unit designed specifically for the personal



and small computer market. The answer/originate unit is FCC approved for handset jack connection with any modular phone. Price: $199. Novation, 18664 Oxnard St., Tarzana, CA 91356.

CIRCLE INQUIRY NO. 148

CALENDAR

Sep 4-6 Personal Computer World Show, Cunard Hotel, Hammersmith, London, computers, peripherals, supplies, software, supporting services, communications, publications for business, home, and educational applications. Timothy Collins, 11, Manchester Sq., London WIE 20Z, 01- 486-1951.

Sep 11-13 Internepcon/Semiconductor International Conference and Exposition, PSA World Trade Center, Republic of Singapore, production machinery, tools, hardware, materials and test instruments keyed to needs of engineering, manufacturing and support personnel of Southeast Asia. Industrial and Scientific Conference Management, 222 W. Adams St., Chicago, IL 60606, (312) 263-4866.

Sep 16-18 Wescon/80, Convention Center, Anaheim, CA, high-technology electronics convention and exhibition with approximately 1200 booths. Robert Myers, 999 N. Sepulveda Blvd., El Segundo, CA 90245, (213) 475-4571.

Sep 19-21 The Decade of Opportunity Home Electronics Show, Brooks Hall, San Francisco, CA, hands-on demos of home electronic products, VTRs, TV games, TVs, radio cassettes, calculators, home computers. Ginger Taylor, J & J Concepts, 5120 Campbell Ave., Suite 208, San Jose, CA 95130, (408) 866-1494.

Sep 22-25 Twelfth Annual Conference of the Society for Management Information Systems, Fairmont Hotel, Philadelphia, PA, examining the need for management information executives to incorporate data processing, word processing, office automation, image processing, telecommunications. SMIS, 111 E. Wacker Dr., Chicago, IL 60601.

Sep 22-25 National Software Package Conference and Exposition, Hyatt Regency, Chicago, IL, recent innovations in systems houses, data processing, telecommunications and word processing. Kim Moloney, Software Info, Professional Exposition Management Co., Suite 545, 222 W. Adams St., Chicago, IL 60606, (312) 263-3131.

Sep 25-26 Ada Introduction and Trends Seminar, Sheraton Motor Inn, Lexington, MA, application examples, lectures, informal sessions on the Ada language as it applies to industry. Prof. Donald French, Institute for Advanced Professional Studies, One Gateway Ctr., Newton, MA 02158, (617) 964-1412.

Sep 26-27 Classroom Applications of Computers in Grades K-12 Conference, Independence High School, San Jose, CA, tutorial sessions, workshops and exhibits of hardware and software. W. Don McKell, Computer-Using Educators, Independence High School, 1776 Educational Park Dr., San Jose, CA 95133.

Sep 27-28 Personal Computer Show and Fleamarket, Holiday Inn North, Newark, NJ, computers, accessories, software, books and parts for hobbyists and businessmen. Kengore Corp., 9 James Ave., Kendall Park, NJ 08824, (201) 297-6918.

Oct 7-10 International Congress for Data Processing and Software Exchange, Intl. Congress Centre Berlin, trends in data processing, development and problems of modern hardware, innovative applications systems, computers and industry and educational tasks in computer age. AMK Berlin, Messedamm 22, D-1000 Berlin 19, W. Germany, (030) 30 38-1.

Oct 8-10 Circulation Computer Systems Symposium, Marriott Hotel, Chicago, IL, selection of hardware and software, new product support, mail room and distribution controls, customer service, subscriber/non-subscriber files, geared for current or potential users of circulation computer systems. American Newspaper Publishers Assoc., Box 17407, Dulles Intl Airport, Washington DC 20041.

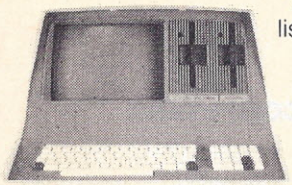
Oct 9-10 Museum Computer Network Annual Conference, New York State Museum, Albany, NY, summarizing the state of computer applications in US and Canadian museums. David Vance, Museum Computer Network, Library E-2340, State U. of NY, Stony Brook, NY 11794, (516) 246-6077.

Oct 8-22 Electronics Tour, Korea, Japan, Taiwan and Hong Kong Electronics Shows, develop foreign markets, observe foreign technology and innovations, seek foreign capital and investment, develop new products and improve personal contacts with foreign counterparts. Commerce Towers Intl, 870 Market St., Suite 762, San Francisco, CA 94102, (415) 433-3072.

Oct 14-16 Mini/Micro Conference and Exposition, Civic Auditorium, Brooks Hall, San Francisco, CA, technical program and product expo devoted to small computers. Mini/Micro Computer Expo, 32302 Camino Capistrano, Suite 202, San Juan Capistrano, CA 92675.

SUPERBRAIN

FUTRA PRICE
$2445
list $2995



WOW! A complete expandable, Z80, S100, CP/M, Computer System. With Dual Drives 350K, 12" CRT, 32K RAM Double Density Superbrain $$ CALL

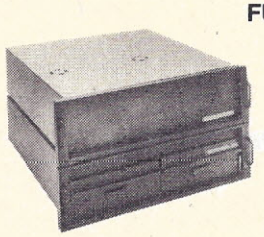
ATARI 800

FUTRA PRICE
$795
list $1085

A 16K RAM personal computing system with BASIC Lang.
Program Recorder $78
Education Cartridge $58

MICROMATION Z+120

FUTRA PRICE
$4145
list $4995



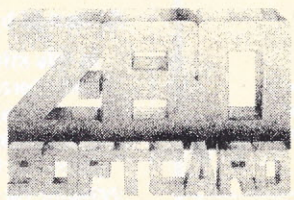
The most expandable 64K Z80, S100 computer System on the market today. With Dual DD 8" Drives.
Z+100 without Drives $2145

MICROMATION S-100 BOARDS

- Z-64 Board (list $1295) **$1099**
  4Mhz Z80 CPU with 64K
- VIO-F8 (list $495) **439**
  80 × 24 Memory Mapped Video with Serial & Parallel I/O
- DS-F8 (list $495) **429**
  Double density controller
- MD Controller (list $895) **769**
  For shugart SA4004-8
- Multi/User I/O (list $295) **249**
  4 Serial, 2 parallel, Clock

FUTRA COMPANY
P. O. BOX 4380
DEPARTMENT I
TORRANCE, CA 90510
Phone (213) 370-2933

MICROSOFT Z-80 Soft Card

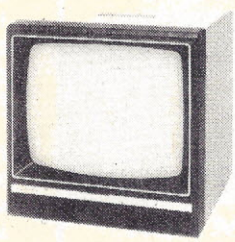


FUTRA PRICE
$259
list $349

Now! Z80-CP/M Capability in your Apple II. Switch between Z80/8080 software and Apple II 6502 software CP/M & Microsoft Basic Incl.

12" SANYO GREEN SCREEN MONITOR

FUTRA PRICE
$274
list $325



800 line resolution monitor with Green Phosphoric Screen for pleasant viewing. Perfect for new 80 by 24 Video Boards
LEEDEX 12" B&W $153

APPLE Accessories

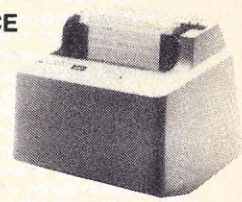
- Novation CAT Modem **$179**
- Apple Clock Card **249**
- ABT Key Pad **115**
- ALF Music Card **239**
- DC Hayes Micromodem II **345**
- Speech Lab Disk Software **189**
- VisiCalc **120**
- CCA Data Management **85**
- Desk Top Plan **85**

DATA SUPPLIES

- 5" ¼" Memorex Diskettes **$27**
  0, 10, or 16 Sector – Box of 10
- 8" Memorex diskettes **$32**
  0 or 32 Sector – Box of 10
- 3 ring Diskette Sleves (10) ... **$6.95**
  Each holds (2) 5" ¼" diskettes

GRAPHICS PAPER TIGER

FUTRA PRICE
$879
list $1095



A high performance dot matrix 96 character printer with Factory Graphics Opt. Serial and Parallel Interface
EMAKO 22 printer $749

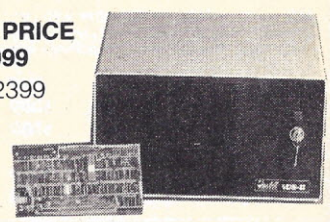
PEACHTREE SOFTWARE

- General Ledger – **$495**
  (list $995)
- Acct. Payable – **$495**
  (list $995)
- Acct. Receivable – **$495**
  (list $995)
- Payroll – **$495**
  (list $995)
- Inventory – **$530**
  (list $1195)

Requires a 8080 or Z80 computer with 48K RAM, CP/M Microsoft Disk BASIC & a 132 column printer

TARBELL VDS-IID

FUTRA PRICE
$2099
list $2399



A complete double density S-100 Disk Subsystem, Dual 8" drives and controller, attractive cabinet, power supply, CP/M, Tarbell BASIC
VDS-IIDD Double Sided $2449

MOUNTAIN HARWARE for Apple II

- Supertalker **$259**
- Romplus with filter **$175**
- Romwriter **$149**
- INTROL X/10 System **$259**

TERMS OF SALE: Master Charge or Visa add 3% service charge to purchase price. No COD's. Allow 3 weeks for personal checks to clear. Orders under $50.00, add $2.00 for shipping and handling costs. All other orders (unless specified in ad) that are within 50 lbs. limit of U.P.S. will be shipped, no charge. Shipments over 50 lbs. in weight or out of Continental U.S. shipped freight collect. All prices are subject to change and all offers subject to withdrawal without notice.
California residents add 6% or 6.5% if serviced by BART.

BOOK REVIEWS

INTRODUCTION TO MICROCOMPUTERS FOR THE HAM SHACK

By Harry L. Helms, Jr.
Howard W. Sams, Indianapolis, IN

Reviewed by Susan Grace

In this informative booklet, Helms celebrates the advent of "computercations"—the introduction of computer technology into electronic communications. His objective is to acquaint the radio amateur with basic microcomputer technology.

Devoted to explaining the five components of a computer system, he starts off describing each component in detail: the CPU, memory, I/O devices and interfaces, and programming. The explanations are basic and specific. He doesn't throw terminology around; he tells what something is and why it has its name. For example, the term "bit" is explained as a contraction of "binary digit;" this may not be important to a computer hobbyist but it is important to a beginner trying to understand the concept.

Subsequently, the author explores microprocessor theories and programming. The programming chapter is easily understood, partly due to flow charts illustrating the programming process. In addition, the brief explanation on computer languages is helpful to the beginner.

The author also discusses the application of microprocessors to amateur radio, such as Morse code transmission and reception, frequency storage and automatic identification. Helms concludes by exploring the future of "computercations," in such forms as medium scan television, packet radio, and a fourth form of digital modulation called pulse code modulation.

The specific explanations, coupled with the illustrations, flow charts, diagrams and glossary, make this an effective beginner's guide to microcomputer concepts in relation to amateur radio applications. □

95 pages, $4.95

MICROCOMPUTER INTERFACING WITH THE 8255 PPI CHIP

By Paul F. Goldsbrough
Howard W. Sams, Indianapolis, IN

Reviewed by Al Baker

The 8255 is a programmable peripheral interface, or PPI. It provides a powerful means for interfacing a microprocessor to the outside world. This book demonstrates just how powerful.

If you can't read assembly language programs, timing diagrams, hardware logic schematics, and don't understand what device select pulses, polled or vectored interrupts, and accumulator versus memory mapped I/O are, start with a more elementary text.

If you are the type of hardware hacker that treats every new integrated circuit as a puzzle created for you to solve, stay as far away as possible. The author leaves nothing to the imagination; you will walk away from this book with a complete understanding of the 8255. On the other hand, if you love to get out your breadboarding equipment, order a bunch of components from one of the mail order houses, and have a great time learning about the latest computer chip, you should proceed.

My biggest complaint about many books published by Howard W. Sams & Co. is that there is no listing of compo-

nents needed to do the experiments. For starters, the reader must have an 8080A based system.

One of the best ways to teach general principles is to use specific examples. The author attempts to do this at two different levels. With over 10 detailed experiments and many examples and explanations, you will learn how to actually use the 8255 PPI. You should then be able to apply the 8255 to more general problems.

The book begins by explaining the many ways that the PPI can be used. Next, you build some of the circuits needed in later experiments. The remainder of the book presents each of the various operating modes of the 8255 and puts them into use. Not only do you build the necessary circuits, but you will spend time writing and using software programs to control the behavior of the PPI.

Two appendices are provided. The first details the electrical and timing characteristics of the 8255, and the second is a single page software summary. The summary would have made an excellent reference card if it had been printed on heavy stock and perforated for easy removal.

Again Sams has produced a book that succeeds at its goal. It is highly recommended. □

217 pages, $8.95

Z-80 MICROPROCESSOR PROGRAMMING AND INTERFACING Books I & II

By Elizabeth A. Nichols, Joseph C. Nichols and Peter R. Rony
Howard W. Sams, Indianapolis, IN

Reviewed by David Marca

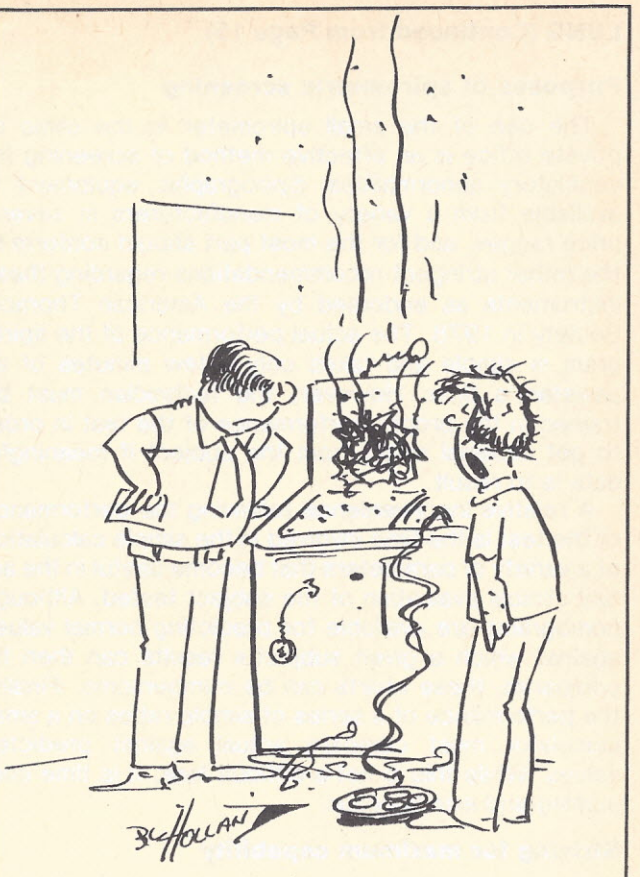
This set of books is one of the best educational products on the market. Both are extremely well written, with objectives for each chapter, to-the-point wording, and topics that build upon one another in an easy-to-follow style. The theme of "learn by doing" is reinforced by the experiment-oriented approach to technical concepts.

Book I covers programming basics, from introductory concepts to a fairly complete investigation of the Z-80 instruction set. All programs are intended for development on the NBZ80 nanocomputer. Adequate space is devoted to familiarizing the reader with its operating essentials. An excellent collection of appendices that organize and summarize Z-80 programming can be a big help when performing experiments.

Book II concentrates on interfacing the NBZ80 nanocomputer to a variety of external devices and circuits. A detailed investigation is done on much of the nanocomputer hardware. Experiments are closely related to those investigations, providing a richer appreciation of the computer's detailed operations. Hundreds of tables, graphs, drawings, and charts provide useful maps of the hardware terrain.

The heavy emphasis on experimentation requires the purchase of the NBZ80 nanocomputer, but it's well worth the investment. While the books are well-suited for organizing the investigation, a notebook and a worktable for experiments are a must. Some basic knowledge of electronics is required, although the books provide much assistance along the way. □

Book I, $10.95; Book II, $12.95



"When'll I be eligible to come back and try out for the job again?"

LUNG (Continued from Page 15)

Purposes of spirometric screening

The use of the small spirometer in the clinic or private office is an effective method of screening for ventilatory abnormalities. Spirographic equipment is available from a variety of manufacturers in several price ranges, and for the most part should conform to the rather stringent recommendations regarding these instruments as endorsed by the American Thoracic Society in 1978. The actual performance of the spirogram is simple and takes only a few minutes of an assistant's time. However, the technician must be trained in the proper performance of the test in order to get maximal effort from the subject if meaningful data is to result.

A relative inconvenience following the performance of the test is the time involved in the simple calculation of a variety of parameters that become useful in the actual clinical evaluation of the subject tested. Although nomograms are available for predicting normal values against which a given subject's results can then be compared, these charts can be cumbersome. Finally, the performance of a series of simple ratios on a small calculator must compare actual against predicted values. While this is not a difficult task, it is time consuming and errors occur.

Striving for maximum capability

Some of the newer models of small spirometers come replete with LED readouts of volumetric data, and so-called computerized versions of the basic instrument are also available. On close inspection, however, it is apparent that these instruments do not preclude a significant amount of hand calculating. Nor is provision made to allow for alterations in the basic program offered, despite the fact that new research on pulmonary medicine prompts changes in normal values and improved statistical analytic techniques. These should be taken into account if the maximum capability of the spirometer is to be realized.

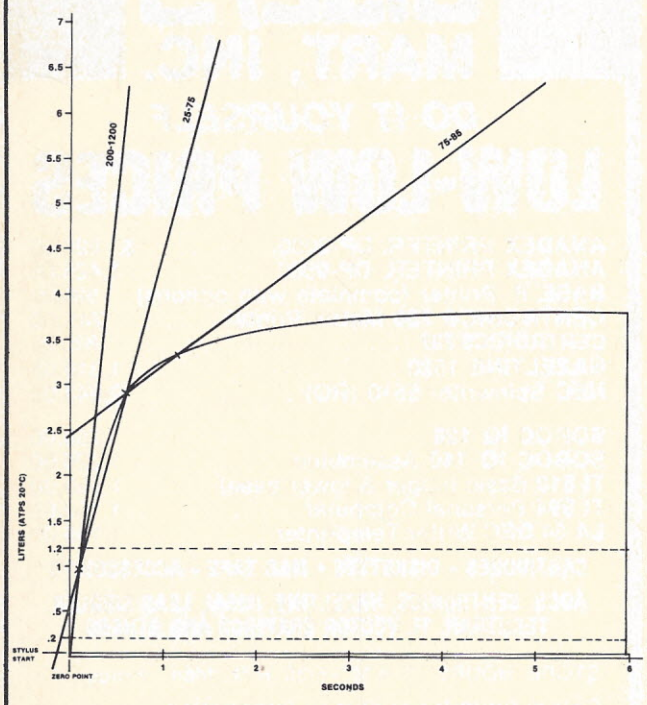


Figure 1. A typical single-breath forced expiratory spirogram. Data derived from this graph as well as from subject's age, sex, and height are employed in the program Lung to evaluate pulmonary volume and various flow characteristics.

```
                    SPIROGRAPHIC ANALYSIS
                    ---------------------

ENTER THE SPECIFIED DATA (NUMBERS ONLY) AS IT IS REQUESTED
-----------------------------------------------------------------

AGE (IN YEARS)? - 40
HEIGHT (IN INCHES)? - 70.5
SEX (M OR F)? - M

INPUT THE FOLLOWING DATA AS CALCULATED FROM THE SPIROGRAM.  USE
NUMERICAL DATA ONLY.  IF DATA IS NOT AVAILABLE, OR WILL NOT BE
NECESSARY, ENTER '0'.
-----------------------------------------------------------------

ACTUAL FVC (BTPS)? - 4.45
ACTUAL FEF 25-75 (ATPS)? - 3.5
ACTUAL FEF 200-1200 (ATPS)? - 9.25
ACTUAL FEF 75-85 (ATPS)? - 1.67
ACTUAL FEV 0.5 (BTPS)? - 3.02
ACTUAL FEV 1 (BTPS)? - 3.6

-----------------------------------------------------------------
-----------------------------------------------------------------
          *  ACTUAL   * PREDICTED * ACTUAL %   * EXPECTED %
          *           *           *    OF     *    OF
          *           *           * PREDICTED * PREDICTED
-----------------------------------------------------------------
FVC       *  4.45     *  5.193    *  85.6923  *    >75
-----------------------------------------------------------------
FEV 1     *  3.6      *  3.946    *  91.2316  *    >75
-----------------------------------------------------------------
FEF       *           *           *           *
  200-1200*  10.1935  *  7.8145   *  130.443   *    >65
-----------------------------------------------------------------
FEF 25-75 *  3.857    *  4.0265   *  95.7904   *    >55
-----------------------------------------------------------------
FEF 75-85 *  1.84034  *  1.2065   *  152.535   *    >75
-----------------------------------------------------------------
-----------------------------------------------------------------
FEV 1/FVC *  80.8989  * >55-70(!) *           *
-----------------------------------------------------------------
FEV 1     *           *           *           *
--------  *  1.19205  *  <1.5     *           *
FEV 0.5   *           *           *           *
-----------------------------------------------------------------
-----------------------------------------------------------------
(!)  EXPECTED % DECREASES WITH AGE

        DO YOU WANT TO DO ANOTHER GRAPH (Y OR N)? - Y
```

Figure 2. Example of graph print-out.

Perhaps the most convincing argument against the purchase of an automated version of the small office spirometer is the cost of the instrument as compared to its non-computerized counterpart. Actually, it is possible to purchase a variety of independent microcomputers with the realized savings in cost between the automated and the non-automated spirometer.

We had made fairly extensive use of a well accepted and dependable spirometer in office practice (the Vitalograph model #20.000). But we had little time to do the necessary calculations to evaluate an increasing number of screening spirograms generated by a three-doctor, primary care clinic. It was decided to utilize a microcomputer for this purpose. The program resides in substantially less than 8K user memory.

An effort was made to keep the working program as simple as possible, since modifications and additions will probably be in order depending on age, race, ethnic or geographic background and other diagnostic parameters. □

PROGRAM LISTING

```
1 REM                    LUNG
2 REM
3 REM     JAMES K. ROBINSON, II, D.O.
4 REM     6505 REVERE AVENUE
5 REM     WAUWATOSA, WISCONSIN  53213
6 REM
7 REM     'LUNG' CALCULATES VARIOUS SIGNIFICANT PULMONARY
8 REM     FUNCTIONAL VOLUME AND FLOW PARAMETERS AS DERIVED
9 REM     FROM DATA OBTAINED FOLLOWING THE PERFORMANCE OF A
10 REM    STANDARD SCREENING SPIROGRAM.
11 REM    THESE RESULTS ARE THEN COMPARED TO PREDICTED
12 REM    'NORMAL' VALUES BASED ON THE LINEAR REGRESSION
13 REM    EQUATIONS DERIVED BY MORRIS, KOSKI, et al.
14 REM
15 REM    USING THE STANDARD ERROR OF ESTIMATE (SEE) VALUE,
16 REM    IT CAN BE DEMONSTRATED THAT 95% OF THE 'NORMAL'
17 REM    POPULATION WILL DEMONSTRATE FUNCTIONAL VALUES
18 REM    EQUAL TO OR GREATER THAN AN 'EXPECTED %' OF
19 REM    PREDICTED' AS INDICATED IN THIS PROGRAM.
20 REM
21 REM    THE REGRESSION EQUATIONS HAVE BEEN CALCULATED
22 REM    FROM RESULTS OBTAINED FROM THE TESTING OF
23 REM    ASYMTOMATIC, NON-SMOKING, CAUCASIAN, AMERICANS,
24 REM    BETWEEN THE AGES OF 20 AND 70.
25 REM    OTHER VALUES MAY BE MORE APPLICABLE TO OTHER
26 REM    AGE RANGES OR OTHER ETHNIC OR RACIAL GROUPS.
100 CLEAR
200 DIM Q(15)
202 DIM S$(50)
203 PRINT TAB(16);"SPIROGRAPHIC ANALYSIS"
205 PRINT:PRINT:PRINT:PRINT
210 PRINT"ENTER THE SPECIFIED DATA (NUMBERS ONLY) AS IT IS REQUE
STED"
212 PRINT"-------------------------------------------------------
------
215 PRINT
220 INPUT "AGE (IN YEARS)";A
225 PRINT
230 IF A<20 OR A>70 THEN GOSUB 2000
240 INPUT "HEIGHT (IN INCHES)";H
245 PRINT
260 INPUT "SEX (M OR F)";S$
275 PRINT:PRINT:PRINT:PRINT
400 PRINT"INPUT THE FOLLOWING DATA AS CALCULATED FROM THE SPIROG
RAM."
410 PRINT"USE NUMERICAL VALUES ONLY.  IF DATA IS NOT AVAILABLE,
420 PRINT"OR WILL NOT BE NECESSARY, ENTER '0'"
422 PRINT"-------------------------------------------------------
------
425 PRINT
430 INPUT "ACTUAL FVC (BTPS)";C2
435 PRINT
440 INPUT "ACTUAL FEF 25-75 (ATPS)";F2
445 PRINT
450 INPUT "ACTUAL FEF 200-1200 (ATPS)";F5
455 PRINT
460 INPUT "ACTUAL FEF 75-85 (ATPS)";F8
465 PRINT
470 INPUT "ACTUAL FEV 0.5 (BTPS)";V0
475 PRINT
480 INPUT "ACTUAL FEV 1 (BTPS)";V3
492 REM     CORRECT ATPS TO BTPS FROM FLOW-RATE CALCULATOR
493 REM     READINGS (VITALOGRAPH)
494 F2=F2*1.102 : F5=F5*1.102 : F8=F8*1.102
495 PRINT:PRINT:PRINT:PRINT
500 C0=(0.115*H)-(0.024*A)-2.852
510 C1=(0.148*H)-(0.025*A)-4.241
520 F0=(0.060*H)-(0.030*A)+0.551
530 F1=(0.047*H)-(0.045*A)+2.513
540 F3=(0.145*H)-(0.036*A)-2.532
550 F4=(0.109*H)-(0.047*A)+2.010
560 F6=(0.025*H)-(0.021*A)+0.321
570 F7=(0.013*H)-(0.023*A)+1.210
580 V1=(0.089*H)-(0.025*A)-1.932
590 V2=(0.092*H)-(0.032*A)-1.260
600 Q(0)=V3/V0
610 Q(1)=(V3/V1)*100
620 Q(2)=(V3/V2)*100
630 Q(3)=(V3/C2)*100
640 Q(4)=(C2/C0)*100
650 Q(5)=(C2/C1)*100
660 Q(6)=(F2/F0)*100
670 Q(7)=(F2/F1)*100
680 Q(8)=(F5/F3)*100
690 Q(9)=(F5/F4)*100
700 Q(10)=(F8/F6)*100
710 Q(11)=(F8/F7)*100
795 REM    810-820 DIFFERENTIATES MALE FROM FEMALE NORMAL VALUES
800 IF S$="F" THEN GOTO 1000
810 C0=C1 : F0=F1 : F3=F4 : F6=F7 : V1=V2 : Q(1)=Q(2)
820 Q(4)=Q(5) : Q(6)=Q(7) : Q(8)=Q(9) : Q(10)=Q(11)
1000 FOR X=1 TO 50
1010 PRINT
1020 NEXT X
1075 REM    1080 BEGINS CHART DISPLAY
1080 GOSUB 4000
1090 GOSUB 4000
1100 PRINT TAB(12);"*  ACTUAL    * PREDICTED *";
1110 PRINT TAB(38);"ACTUAL %  * EXPECTED %"
1120 PRINT TAB(12);"*"; TAB(24);"*"; TAB(36);"*";
1130 PRINT TAB(41);"OF      *    OF"
1140 PRINT TAB(12);"*"; TAB(24);"*"; TAB(36);"*";
1150 PRINT TAB(38);"PREDICTED * PREDICTED"
1160 GOSUB 4000
1180 PRINT"FVC"; TAB(12);"* ";C2; TAB(24);"* ";C0;
1190 PRINT TAB(36);"* ";Q(4); TAB(48);"*    >75"
1200 GOSUB 4000
1240 PRINT"FEV 1"; TAB(12);"* ";V3; TAB(24);"* ";
1250 PRINT TAB(26);V1; TAB(36);"* ";Q(1); TAB(48);"*    >75"
1260 GOSUB 4000
1270 PRINT"FEF"; TAB(12);"*"; TAB(24);"*"; TAB(36);"*"; TAB(48);
"*"
1280 PRINT"  200-1200"; TAB(12);"* ";F5; TAB(24);"* ";F3;
1290 PRINT TAB(36);"* ";Q(8); TAB(48);"*    >65"
1300 GOSUB 4000
1310 PRINT"FEF 25-75"; TAB(12);"* ";F2; TAB(24);"* ";F0;
1320 PRINT TAB(36);"* ";Q(6); TAB(48);"*    >55"
1330 GOSUB 4000
1340 PRINT"FEF 75-85"; TAB(12);"* ";F8; TAB(24);"* ";F6;
1350 PRINT TAB(36);"* ";Q(10); TAB(48);"*    >75"
1360 GOSUB 4000
1365 GOSUB 4000
1370 PRINT"FEV 1/FVC"; TAB(12);"* ";Q(3); TAB(24);"*";
1375 PRINT TAB(26);">55-70(!)";
1380 PRINT TAB(36);"*"; TAB(48);"*"
1390 GOSUB 4000
1400 PRINT"FEV 1"; TAB(12);"*"; TAB(24);"*"; TAB(36);"*";
1410 PRINT TAB(48);"*"
1420 PRINT"--------"; TAB(12);"* ";Q(0); TAB(24);"* <1.5";
1425 PRINT TAB(36);"*"; TAB(48);"*"
1430 PRINT"FEV 0.5"; TAB(12);"*"; TAB(24);"*";
1440 PRINT TAB(36);"*"; TAB(48);"*"
1450 GOSUB 4000
1460 GOSUB 4000
1470 PRINT"(!)  EXPECTED % DECREASES WITH AGE"
1475 REM    1470 ENDS CHART DISPLAY
1500 PRINT:PRINT
1510 INPUT "DO YOU WISH TO DO ANOTHER GRAPH (Y OR N)";X$
1520 IF X$="N" THEN GOTO 5000
1530 FOR X=1 TO 50
1540 PRINT
1550 NEXT X
1560 GOTO 100
1995 REM    SUB-ROUTINES FOLLOW
2000 PRINT "THESE CALCULATIONS ARE NOT VALID FOR PERSONS UNDER"
2010 PRINT "THE AGE OF 20 NOR OVER THE AGE OF 70 YEARS."
2020 PRINT
2030 RETURN
4000 PRINT"--------------------------------------";
4010 PRINT"-----------------------"
4020 RETURN
5000 END
READY
```

PURCHASE ORDER (Continued from Page 68)

PROGRAM LISTING

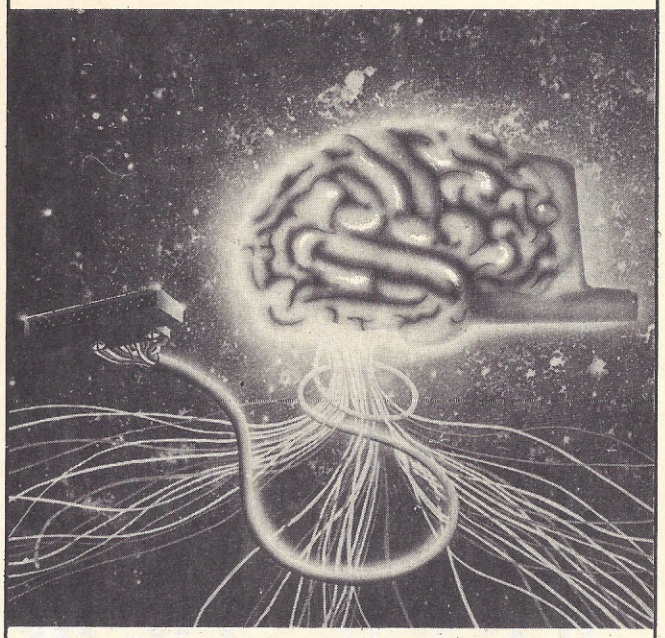
```
0100 DEF FNA(X)=36-(X/2)
0110 DEF FNB(Y)=48-LEN(STR$(Y))
0120 DEF FNC(Z)=62-LEN(STR$(Z))
0130 POKE( 62,50)
0140 INPUT "P.O. NUMBER",A$(7)
0150  A$(7)="P.O. NR: "+A$(7)
0160 LINE= 0
0200 INPUT "NAME",A$(1)
0210 INPUT "STREET ADDRESS",A$(2)
0220 INPUT "CITY, STATE  ZIP",A$(3),A$(4)
0230 PRINT
0240 INPUT "TODAY'S DATE (D-M-Y)",D$
0250 INPUT "TO",B$(1)
0260 INPUT "STREET ADDRESS",B$(2)
0270 INPUT "CITY, STATE  ZIP",B$(3),B$(4)
0280 PRINT
0290 INPUT "HOW MANY ITEMS",N
0300 PRINT
0310 FOR J=1 TO N
0320 INPUT "QUANTITY",Q(J)
0330 INPUT "DESCRIPTION",D$(J)
0340 INPUT "UNIT PRICE",P(J)
0350 NEXT J
0360 PRINT
0370 INPUT "SHIPPING & HANDLING ALLOWANCE",S
0380 INPUT "FOR RESALE",Z$
0390 IF LEFT$(Z$,1)<>"Y" THEN 410
0400  A=1:INPUT"RESALE NUMBER",A$(5)
0410  A$(5)="RESALE TAX NR: "+A$(5)
0420 INPUT "AUTHORIZING SIGNATURE",A$(6)
0430 INPUT "WHAT OUTPUT PORT",Z
0440 INPUT "PRESS 'RETURN' TO PRINT",Z$
0450 PORT= Z
0460 PRINT :PRINT:PRINT
0470 PRINT TAB(29);"PURCHASE ORDER":PRINT:PRINT
0480 PRINT A$(7);
0485 IF A=0 THEN 500
0490 PRINT TAB(71-LEN(A$(5)));A$(5)
0500 PRINT :PRINT
0510 PRINT TAB(FNA(LEN(A$(1))));A$(1)
0520 PRINT TAB(FNA(LEN(A$(2))));A$(2)
0530 PRINT TAB(FNA(LEN(A$(3)+A$(4))+2));A$(3);", ";A$(4)
0540 PRINT
0550 PRINT TAB(FNA(LEN(D$)));D$
0560 PRINT :PRINT:PRINT:PRINT
0570 PRINT "TO:";
0580 PRINT TAB(5);B$(1)
0590 PRINT TAB(5);B$(2)
0600 PRINT TAB(5);B$(3);", ";B$(4)
0610 PRINT
0620 PRINT "PLEASE SHIP THE FOLLOWING TO US:"
0630 PRINT
0640 PRINT "QTY";
0650 PRINT TAB(15);"DESCRIPTION";
0660 PRINT TAB(40);"UNIT PRICE";
0670 PRINT TAB(55);"EXTENSION"
0680 PRINT
0690  T=0
0700 FOR J=1 TO N
0710 DIGITS= 0
0720 PRINT Q(J);
0730 PRINT TAB(10);D$(J);
0740 DIGITS= 2
0750 PRINT TAB(FNB(P(J)));P(J);
0760  P=Q(J)*P(J)
0770 PRINT TAB(FNC(P));P
0780  T=T+P
0790 PRINT
0800 NEXT J
0810 PRINT TAB(53);"----------"
0820 PRINT TAB(5);"TOTAL FOR MERCHANDISE";
0830 PRINT TAB(60-LEN(STR$(T)));"$ ";T
0840 PRINT :PRINTTAB(5);"ALLOWANCE FOR SHIPPING/HANDLING";
0850 PRINT TAB(FNC(S));S
0860 IF A=1 THEN 880
0865  X=.06*T
0870 PRINT :PRINT"SALES TAX";TAB(FNC(X));X
0880 PRINT TAB(53);"----------"
0890 PRINT "TOTAL (ENCLOSED)";
0900 PRINT TAB(60-LEN(STR$(T+S+X)));"$ ";S+T+X
0910 PRINT TAB(53);"=========="
0920 PRINT :PRINTTAB(5);"THANK YOU!"
0930 PRINT :PRINT:PRINT:PRINT
0940 PRINT TAB(35);"------------------------"
0950 PRINT TAB(47-(LEN(A$(6)))/2);A$(6)
0960 PRINT TAB(47-(LEN(A$(1)))/2));A$(1)
0970 PORT= 1
0980 INPUT "ANOTHER COPY",Z$
0990 IF LEFT$(Z$,1)="Y" THEN 430
0999 END
```

Sample Run

```
P.O. NUMBER? 80-112
NAME? BUD'S COMPUTER WORKS
STREET ADDRESS? #1 SOFTWARE LANE
CITY, STATE  ZIP? UTOPIA, CA 99999
TODAY'S DATE (D-M-Y)? 20 MAY 1980
TO? COMPUTER SUPPLIES COMPANY
STREET ADDRESS? 123 MAIN STREET
CITY, STATE  ZIP? CORNUCOPIA, NY 11111
HOW MANY ITEMS? 5
QUANTITY? 10
DESCRIPTION? 10 SECTOR MINIDISK
UNIT PRICE? 3.5
QUANTITY? 1
DESCRIPTION? 8 1/2 INCH PAPER ROLL
UNIT PRICE? 8.40
QUANTITY? 12
DESCRIPTION? TTY RIBBON
UNIT PRICE? 2.25
QUANTITY? 2
DESCRIPTION? DISKETTE HOLDER
UNIT PRICE? 5
QUANTITY? 1
DESCRIPTION? DISKETTE ERASER
UNIT PRICE? 34.69

SHIPPING & HANDLING ALLOWANCE? 12
FOR RESALE? Y
RESALE NUMBER? 12345XTX
AUTHORIZING SIGNATURE? I. M. OUNER
WHAT OUTPUT PORT? 3
PRESS 'RETURN' TO PRINT?
```

NERVE SIMULATION (Continued from Page 74)

PROGRAM LISTING



```
10  REM  ---> HODGKIN-HUXLEY NERVE SIMULATION <---

500  REM  -> GET STIM; INITIALIZE <-
510  INPUT "STIMULUS AMPLITUDE = ";SA
520  INPUT "STIMULUS DURATION  = ";SD
530 MV =  - 90:T = 0:DT = 1 / 25
540  HGR2 : REM    CLEAR SCREEN, HIRES GRAPHICS
550  GOSUB 2000: REM   SET ALPHA,BETA
560  GOSUB 2500: REM   SET INITIAL N,M,H
570  REM

1000  REM          MAIN LOOP IS HERE  <--------------
1010  GOSUB 2000: REM  SET ALPHA,BETA FOR MV
1020  GOSUB 4000: REM  UPDATE VARIABLES
1030  GOSUB 5000: REM  PLOT VARIABLES
1040 T = T + DT: IF T < 10 THEN  GOTO 1000: REM --->
1050  END
1060  REM

2000  REM  -> SET ALPHA, BETA FOR MV <-
2010 V =  - MV - 90: REM  OFFSET SO V=0 AT REST
2020 BN = 0.125 *  EXP (V / 80)
2030 BM = 4 *  EXP (V / 18)
2040 BH = 1 / ( EXP ((V + 30) / 10) + 1)
2050 AN = .01 * (V + 10) / ( EXP ((V + 10) / 10) - 1)
2060 AM = 0.1 * (V + 25) / ( EXP ((V + 25) / 10) - 1)
2070 AH = 0.07 *  EXP (V / 20)
2080  RETURN : REM

2500  REM  -> INITIAL DIMENSIONLESS N, M, H <-
2510 N = AN / (AN + BN)
2520 M = AM / (AM + BM)
2530 H = AH / (AH + BH)
2540  RETURN : REM

4000  REM  -> UPDATE VARIABLES <-
4010 DN = AN * (1 - N) - BN * N:N = N + DN * DT
4020 DM = AM * (1 - M) - BM * M:M = M + DM * DT
4030 DH = AH * (1 - H) - BH * H:H = H + DH * DT
4040 GK = 36 * N * N * N * N
4050 GNA = 120 * M * M * M * H
4060 IK = GK * (V - 12):INA = GNA * (V + 115)
4070 IL = 0.3 * (V + 10.6):IT = IK + INA + IL
4080  IF T <  = SD THEN IT = IT + SA: REM STIM?
4090 V = V - IT * DT:MV =  - V - 90
4100  RETURN : REM

5000  REM  -> PLOT MV AND CONDUCTANCES <-
5005 X = 25 + 250 * (T / 10)
5010  HPLOT X,20 - (MV / 1.5)
5030  HPLOT X,150 - (1.5 * GNA)
5040  HPLOT X,150 - (1.5 * GK)
5050  RETURN : REM
```

BEATING THE SYSTEM (Continued from Page 78)

PROGRAM LISTING

Source Code for Hexedit Program

```
PROGRAM HEXEDIT;

(*************************************)
(*                                   *)
(*   HEXADECIMAL FILE EDITOR         *)
(*   MARK BORGERSON                  *)
(*   THE COMPUTER STORE OF CORVALLIS  *)
(*   2015 NW CIRCLE BLVD.            *)
(*   CORVALLIS OR.  97330            *)
(*   JANUARY 1980                    *)
(*************************************)

USES APPLESTUFF;
(* ONLY THE KEYPRESS FUNCTION IS USED *)


TYPE
      NYBBLE=0..15;
      BYTE=0..255;
      WORD=PACKED ARRAY[1..2]OF BYTE;
      SETOFCHAR=SET OF CHAR;


VAR
      DATFILE:FILE OF WORD;
      OUTDEV:FILE OF CHAR;
      CSTRING:STRING;
      DATWORD:WORD;
      FILNAME:STRING[40];
      HEXSTRING:STRING[16];
      OPTION,STOPCHAR:CHAR;
      FILADR:WORD;


PROCEDURE PRBYTE(ABYTE:BYTE);
(* PRINT A BYTE AS TWO HEX CHARACTERS *)
VAR NYB1,NYB2:NYBBLE;

BEGIN
  NYB1:=ABYTE MOD 16;
  NYB2:=ABYTE DIV 16;
  WRITE(OUTDEV,HEXSTRING[NYB2+1],HEXSTRING[NYB1+1]);
END; (*PRBYTE*)

FUNCTION MAKEINT(AWORD:WORD):INTEGER;
(* MAKE A 16-BIT WORD INTO A POSITIVE INTEGER *)
(* FOR USE IN ACCESSING THE DATA FILE         *)

BEGIN
  REPEAT
    READ(KEYBOARD,CH);
    IF EOLN(KEYBOARD) THEN CH:=CHR(13);
    GOOD:= CH IN OKSET;
    IF NOT GOOD THEN WRITE(CHR(7))
      ELSE IF CH IN [' '..'}'] THEN WRITE(CH);
  UNTIL GOOD;
  GETCHAR:=CH;
END;


PROCEDURE HEXIN(VAR S: STRING; MAXLEN: INTEGER);
(*************************************************************)
(*                                                           *)
(*  GET AND ECHO A STRING UP TO MAXLEN CHARS LONG.           *)
(*  IF NULL STRING ENTERED, DEFAULT AND PRINT PREVIOUS VALUE.*)
(*  THIS ROUTINE ALSO FROM THE "DISKIO" PROGRAM              *)
(*                                                           *)
(*************************************************************)
VAR S1: STRING[1];
    STEMP: STRING[8];
    OKSET: SET OF CHAR;
BEGIN
  OKSET:=[CHR(27),'0'..'9','A'..'F'];
  S1:=' ';
  STEMP:='';
  REPEAT
    IF LENGTH(STEMP) = 0 THEN S1[1]:=GETCHAR(OKSET + [CHR(13)])
      ELSE IF LENGTH(STEMP)=MAXLEN THEN S1[1]:=GETCHAR([CHR(13),CHR(8)])
            ELSE S1[1]:=GETCHAR(OKSET + [CHR(13),CHR(8)]);
    IF S1[1] IN OKSET THEN STEMP:=CONCAT(STEMP,S1)
      ELSE IF S1[1]=CHR(8) THEN
        BEGIN
          WRITE(S1[1]);
          DELETE(STEMP,LENGTH(STEMP),1);
        END;
  UNTIL (S1[1] = CHR(13))OR(S1[1]=CHR(27));
  IF LENGTH(STEMP) <> 0 THEN S:=STEMP
  ELSE WRITE(S);
END;


PROCEDURE GETWORD(VAR AWORD:WORD;
                  VAR ESCP:BOOLEAN);
(*  GET A 16-BIT WORD FOR EITHER ADDRESS OR DATA   *)
(*  IF NULL STRING ENTERED, KEEP OLD VALUE OF AWORD*)
(*  ESCP VARIABLE USED TO TEST FOR EXIT FROM MODIFY*)

VAR WORDSTRING:STRING[4];
    NYBSTRING:STRING[1];
    WD1,WD2,HEXVAL:INTEGER[6];
    I:INTEGER;
```

```
BEGIN
  IF AWORD[1]>127 THEN MAKEINT:=0
  ELSE MAKEINT:=AWORD[1]*256+AWORD[2];
END;

PROCEDURE INCWORD(VAR AWORD:WORD);
(* INCREMENT A 16-BIT WORD--USED AS AN ADDRESS*)
BEGIN
  IF AWORD[2]=255 THEN AWORD[2]:=0
    ELSE AWORD[2]:=AWORD[2]+1;
  IF (AWORD[2]=0)AND(AWORD[1]<>255) THEN
      AWORD[1]:=AWORD[1]+1;
  IF (AWORD[2]=0) AND (AWORD[1]=255) THEN
      AWORD[1]:=0;
END;

PROCEDURE ADRPRINT(ADDRESS:WORD);
VAR HBYTE,LBYTE:BYTE;

BEGIN
  HBYTE:=ADDRESS[1];
  LBYTE:=ADDRESS[2];
  PRBYTE(HBYTE);
  PRBYTE(LBYTE);
  WRITE(OUTDEV,': ');
END; (*ADRPRINT*)

PROCEDURE ENDLINE;
(* END UP A LINE OF DATA BY PRINTING ASCII CHARACTERS*)
BEGIN
  WRITELN(OUTDEV,'  ',CSTRING);
  CSTRING:='';
  ADRPRINT(FILADR);
END;



PROCEDURE DELAY(DTIME:INTEGER);
  VAR I:INTEGER;
BEGIN
  FOR I:=1 TO DTIME DO;
END;



FUNCTION GETCHAR(OKSET: SETOFCHAR): CHAR;
(*****************************************************************)
(*                                                               *)
(*  GET A CHARACTER, BEEP IF NOT IN OKSET, ECHO ONLY IF PRINTING *)
(*  THIS ROUTINE FROM THE "DISKIO" PROGRAM PROVIDED BY APPLE     *)
(*                                                               *)
(*****************************************************************)
VAR CH: CHAR;
    GOOD: BOOLEAN;
```

```
BEGIN
  ESCP:=FALSE;
  HEXVAL:=0;
  WORDSTRING:='';
  NYBSTRING:='0';
  HEXIN(WORDSTRING,4);
  IF LENGTH(WORDSTRING)<>0 THEN
    IF WORDSTRING[LENGTH(WORDSTRING)]<>CHR(27) THEN
      BEGIN
        FOR I:=1 TO LENGTH(WORDSTRING) DO
          BEGIN
            NYBSTRING[1]:=WORDSTRING[I];
            HEXVAL:=HEXVAL*16+POS(NYBSTRING,HEXSTRING)-1;
          END;
        WD1:=HEXVAL DIV 256;
        WD2:=HEXVAL-WD1*256;
        AWORD[1]:=TRUNC(WD1);
        AWORD[2]:=TRUNC(WD2);
        WRITELN;
      END
    ELSE ESCP:=TRUE;
END; (*GETWORD*)


PROCEDURE DUMP;
VAR   BTSTRING:STRING;
      CNTRLP,CNTRLQ,ESCAPE,SPACE:CHAR;
      STARTAT,DATBYTE,CVAL,WIDTH:INTEGER;
      ESCP:BOOLEAN;
(***********************************************)
(*  DUMP DATA IN HEXADECIMAL AND ASCII FORMAT  *)
(***********************************************)
BEGIN
  CNTRLP:=CHR(16);
  CNTRLQ:=CHR(17);
  ESCAPE:=CHR(27);
  SPACE:=' ';
  PAGE(OUTPUT);
  WRITE('STARTING ADDRESS (HEX): ');
  REWRITE(OUTDEV,'CONSOLE:');
  FILADR[1]:=0;
  FILADR[2]:=0;
  ESCP:=FALSE;
  GETWORD(FILADR,ESCP);
  STARTAT:=MAKEINT(FILADR);
  SEEK(DATFILE,STARTAT DIV 2);
  GET(DATFILE);
  PAGE(OUTPUT);
  (* ADD A "READLN(WIDTH) ETC. HERE TO MODIFY OUPUT FORMAT*)
  WIDTH:=8;
  CSTRING:='';
  STOPCHAR:=SPACE;
  BTSTRING:=' ';
  WRITELN(OUTDEV);
  ADRPRINT(FILADR);
```

```
    WHILE(NOT EOF(DATFILE))AND(STOPCHAR<>ESCAPE ) DO
      BEGIN
        IF KEYPRESS THEN READ(KEYBOARD,STOPCHAR);
        (* TEST FOR SWITCH TO PRINTER *)
        IF STOPCHAR=CNTRLP  THEN
          BEGIN
            STOPCHAR:=SPACE;
            WRITELN(OUTDEV);
            CLOSE(OUTDEV);
            REWRITE(OUTDEV,'PRINTER:');
          END;
        IF STOPCHAR=CNTRLQ  THEN
          BEGIN
            STOPCHAR:=SPACE;
            WRITELN(OUTDEV);
            CLOSE(OUTDEV);
            REWRITE(OUTDEV,'CONSOLE:');
          END;
        DATBYTE:=DATFILE^[1];
        PRBYTE(DATBYTE);
        CVAL:=DATBYTE;
        IF (CVAL<32)OR(CVAL>127)THEN CVAL:=46;
        BTSTRING[1]:=CHR(CVAL);
        CSTRING:=CONCAT(CSTRING,BTSTRING);
        WRITE(OUTDEV,' ');
        INCWORD(FILADR);
        DATBYTE:=DATFILE^[2];
        PRBYTE(DATBYTE);
        CVAL:=DATBYTE;
        IF (CVAL<32)OR(CVAL>127)THEN CVAL:=46;
        BTSTRING[1]:=CHR(CVAL);
        CSTRING:=CONCAT(CSTRING,BTSTRING);
        WRITE(OUTDEV,SPACE);
        INCWORD(FILADR);
        GET(DATFILE);
        IF (MAKEINT(FILADR) MOD WIDTH)=0 THEN ENDLINE;
      END;
    CLOSE(OUTDEV);
  END; (*DUMP*)

  PROCEDURE MODIFY;
  VAR ESCP:BOOLEAN;
      DATBYTE:BYTE;
      STARTAT:INTEGER;
  (*******************************************************)
  (* MODIFY 16-BIT WORDS STARTING AT REQUESTED ADDRESS *)
  (*******************************************************)

  BEGIN
    WRITELN;
    REWRITE(OUTDEV,'CONSOLE:');
    WRITE('STARTING ADDRESS: ');
    ESCP:=FALSE;
    FILADR[1]:=0;
    FILADR[2]:=0;
    GETWORD(FILADR,ESCP);
    STARTAT:=MAKEINT(FILADR) DIV 2;
    SEEK(DATFILE,STARTAT);
    REPEAT
      WRITELN;
      ADRPRINT(FILADR);
      GET(DATFILE);
      DATWORD:=DATFILE^;
      PRBYTE(DATWORD[1]);
      PRBYTE(DATWORD[2]);
      WRITE('  ');
      GETWORD(DATWORD,ESCP);
      SEEK(DATFILE,STARTAT);
      DATFILE^:=DATWORD;
      PUT(DATFILE);
      STARTAT:=STARTAT+1;
      INCWORD(FILADR);
      INCWORD(FILADR);
    UNTIL ESCP;
    CLOSE(OUTDEV);
  END; (*MODIFY*)


  (************************************)
  (*  FILE PATCHING UTILITY           *)
  (*  MAIN PROGRAM                    *)
  (*  JANUARY, 1980                   *)
  (************************************)


  BEGIN
    HEXSTRING:='0123456789ABCDEF';
    PAGE(OUTPUT);
    GOTOXY(0,4);
    WRITELN('FILE EDITING UTILITY');
    WRITELN;
    WRITE('FILE NAME: ');
    READLN(FILNAME);
    RESET(DATFILE,FILNAME);
    WRITELN;
    WRITELN('YOU MAY CHOOSE ONE OF THE FOLLOWING: ');
    WRITELN;
    WRITELN('  D: DUMP IN HEX AND ASCII');
    WRITELN('  M: MODIFY FILE CONTENTS ');
    WRITELN('  E: END PROGRAM');
    REPEAT
      WRITELN;
      WRITE('OPTION: ');
      OPTION:=GETCHAR(['D','M','E']);
      IF OPTION='D' THEN DUMP;
      IF OPTION='M' THEN MODIFY;
    UNTIL OPTION='E';
    CLOSE(DATFILE);
  END.
```

THE ORIGINAL MAGAZINE FOR OWNERS OF THE TRS-80™* MICROCOMPUTER

OFTWARE
R TRS-80™
OWNERS

H & E COMPUTRONICS INC.

MONTHLY
NEWSMAGAZIN
FOR TRS-80™
OWNERS

MONTHLY NEWSMAGAZINE

Practical Support For Model I & II

- PRACTICAL APPLICATIONS
- BUSINESS
- GAMBLING • GAMES
- EDUCATION
- PERSONAL FINANCE
- BEGINNER'S CORNER
- NEW PRODUCTS
- SOFTWARE EXCHANGE
- MARKET PLACE
- QUESTIONS AND ANSWERS
- PROGRAM PRINTOUTS

. . . . AND MORE

PROGRAMS AND ARTICLES PUBLISHED IN OUR FIRST 12 ISSUES INCLUDE THE FOLLOWING:

- A COMPLETE INCOME TAX PROGRAM (LONG AND SHORT FORM)
- INVENTORY CONTROL
- STOCK MARKET ANALYSIS
- WORD PROCESSING PROGRAM (FOR DISK OR CASSETTE)
- LOWER CASE MODIFICATION FOR YOUR VIDEO MONITOR OR PRINTE
- PAYROLL (FEDERAL TAX WITHHOLDING PROGRAM)
- EXTEND 16-DIGIT ACCURACY TO TRS-80™ FUNCTIONS (SUCH AS SQUARE ROOTS AND TRIGONOMETRIC FUNCTIONS)
- NEW DISK DRIVES FOR YOUR TRS-80™
- PRINTER OPTIONS AVAILABLE FOR YOUR TRS-80™
- A HORSE SELECTION SYSTEM***ARITHMETIC TEACHER
- COMPLETE MAILING LIST PROGRAMS (BOTH FOR DISK OR CASSETT SEQUENTIAL AND RANDOM ACCESS)
- RANDOM SAMPLING***BAR GRAPH
- CHECKBOOK MAINTENANCE PROGRAM
- LEVEL II UPDATES***LEVEL II INDEX
- CREDIT CARD INFORMATION STORAGE FILE
- BEGINNER'S GUIDE TO MACHINE LANGUAGE AND ASSEMBLY LANGUAGE
- LINE RENUMBERING
- AND CASSETTE TIPS, PROGRAM HINTS, LATEST PRODUCTS COMING SOON (GENERAL LEDGER, ACCOUNTS PAYABLE AND RECEIVABLE, FORTRAN-80, FINANCIAL APPLICATIONS PACKAGE PROGRAMS FOR HOMEOWNERS, MERGE TWO PROGRAMS, STATISTICAL AND MATHEMATICAL PROGRAMS (BOTH ELEMENTARY AND ADVANCED) . . . AND

FREE

WORD PROCESSING PROGRAM (Cassette or Disk) For writing letters, text, mailing lists, etc., with each new subscriptions or renew

LEVEL II RAM TEST (Cassette or Disk) Checks random access memory to ensure that all memory locations are working properly.

DATA MANAGEMENT SYSTEM (Cassette or Disk) Complete file management for your TRS-80™

CLEANUP (Cassette or Disk) Fast action Maze Game

ADVENTURE (Cassette or Disk) Adventure #0 by Scott Adams (From Adventureland International)

FREE

* TRS-80™ IS A TRADEMARK OF TANDY COR

SEND FOR OUR NEW 48 PAGE SOFTWARE CATALOG (INCLUDING LISTINGS OF HUNDREDS OF TRS-80™ PROGRAMS AVAILABLE ON CASSETTE AND DISKETTE). $2.00 OR **FREE** WITH EACH SUBSCRIPTIONS OR SAMPLE ISSUE.

CIRCLE INQUIRY NO. 87

COMPUTRONICS INC.
MATHEMATICAL APPLICATIONS SERVICE™

50 N. PASCACK ROAD
SPRING VALLEY, NEW YORK 10977

master charge

24 HOUR ORDER LINE
(914) 425-1535

VISA

NEW TOLL-FREE ORDER LINE
(OUTSIDE OF N.Y. STATE)
(800) 431-2818

ONE YEAR SUBSCRIPTION $24 .....
TWO YEAR SUBSCRIPTION $48 .....
SAMPLE OF LATEST ISSUE $ 4 .....
START MY SUBSCRIPTION WITH ISSUE .....
(#1 - July 1978 • #7 - January 1979 • #12 - June 1979 • #18 - January 1980)
NEW SUBSCRIPTION ..... RENEWAL .....

NEW!!!
MOD-II NEWSLETTER
$12/year (or 12 issues)

CREDIT CARD NUMBER _____ EXP. DATE _____

SIGNATURE _____

NAME _____

ADDRESS _____ CITY _____ STATE _____ ZIP _____

*** ADD $6/YEAR (CANADA, MEXICO) - ADD $12/YEAR AIR MAIL - OUTSIDE OF U.S.A., CANADA & MEXICO ***

DAMAGED DISK (Continued from Page 86)

```
                ;
                ;  CONVERTS BLOCK NUMBER TO TRACK AND INTERLACED SECTOR
                                                              NUMBER
                ;  BLOCK NUMBER IN REG B - TRACK/SECTOR RETURNED IN H,L
0260 C5         CNVRTB  PUSH    B
0261 68                 MOV     L,B     ;BLOCK TO H,L
0262 2600               MVI     H,0
0264 29                 DAD     H       ;*2
0265 29                 DAD     H       ;*4
0266 29                 DAD     H       ;*8
0267 110002             LXI     D,DBASE*256 ;BASE TRACK NUMBER
026A 01E6FF             LXI     B,-SECTS ;DIVIDE BY SECTORS PER TRACK
026D 7C         CNVRTC  MOV     A,H     ;OVER SECTS?
026E B7                 ORA     A
026F C27802             JNZ     CNVRTT  ;BY GROUPS
0272 7D                 MOV     A,L     ;OVER SECTS?
0273 FE1A               CPI     SECTS
0275 DA7D02             JC      CNVRTS  ;DOWN TO TRACK
0278 09         CNVRTT  DAD     B       ;TAKE AWAY SECTS
0279 14                 INR     D       ;ADD 1 TO TRACK NUMBER
027A C36D02             JMP     CNVRTC  ;GO BACK FOR MORE
027D 5D         CNVRTS  MOV     E,L     ;RESIDUAL=INTERLACED SECTOR-1
027E 1C                 INR     E       ;BUMP FOR 1-26
027F EB                 XCHG            ;PUT TRACK/SECTOR IN H,L
0280 C1                 POP     B
0281 C9                 RET
                ;
                ;  READS A LOGICAL SECTOR (IF IT CAN) RETURNS ZERO
                                                          FLAG SET IF OK
0282 C5         READS   PUSH    B
0283 E5                 PUSH    H
0284 CDA302             CALL    ITOA    ;CONVERT LOGICAL TO ACTUAL SECTOR
0287 E5                 PUSH    H       ;SAVE SECTOR NUMBER
0288 4C                 MOV     C,H     ;TRACK NUMBER IN H
0289 CD8902     SETTRK  CALL    SETTRK  ;BIOS TRACK SET (FIXED BY IBIOS)
028C C1                 POP     B       ;PUT SECTOR IN C
028D CD8D02     SETSEC  CALL    SETSEC  ;BIOS SECTOR SET (FIXED BY IBIOS)
0290 CD9002     DREAD   CALL    DREAD   ;BIOS SECTOR READ (FIXED BY IBIOS)
0293 B7                 ORA     A       ;CHECK FOR READ ERROR
0294 E1                 POP     H
0295 C1                 POP     B
0296 F5                 PUSH    PSW
0297 2C                 INR     L       ;BUMP TO NEXT SECTOR
0298 7D                 MOV     A,L
0299 FE1B               CPI     SECTS+1 ;CHECK FOR TRACK OVERFLOW
029B DAA102             JC      READSR
029E 2E01               MVI     L,1     ;RESET SECTOR TO 1
02A0 24                 INR     H       ; AND BUMP TRACK
02A1 F1         READSR  POP     PSW
02A2 C9                 RET
                ;
                ;  CONVERT LOGICAL (INTERLACED) SECTOR TO ACTUAL
                                                        (PHYSICAL) SECTOR
02A3 EB         ITOA    XCHG
02A4 01AD02             LXI     B,LPMAP-1 ;BASE OF MAP
02A7 6B                 MOV     L,E
02A8 2600               MVI     H,0     ;LOGICAL SECTOR OFFSET
02AA 09                 DAD     B
02AB 5E                 MOV     E,M     ;GET PHYSICAL SECTOR
02AC EB                 XCHG            ;PUT H,L BACK
02AD C9                 RET
```

```
035B 1A         SETDML  LDAX    D
035C 77                 MOV     M,A
035D 13                 INX     D
035E 23                 INX     H
035F 05                 DCR     B
0360 C25B03             JNZ     SETDML
0363 C9                 RET
                ;
0364 110C04     CLOSEB  LXI     D,BFCB  ;FCB FOR [UNUSED].BAD
0367 0E10               MVI     C,16    ;CLOSE FILE
0369 CD0500             CALL    BDOS
036C C9                 RET
                ;
                ;  CONVERT NUMBER OF BLOCKS TO DECIMAL ASCII FOR PRINTING
036D 2A4B04     SETNUM  LHLD    DMCNT   ;GET NUMBER OF SECTORS
0370 29                 DAD     H
0371 29                 DAD     H       ;MAKE H=NUMBER OF BLOCKS
0372 29                 DAD     H
0373 29                 DAD     H
0374 29                 DAD     H
0375 11FF00             LXI     D,255
0378 19                 DAD     D       ;ROUND UP
0379 6C                 MOV     L,H
037A 2600               MVI     H,0     ;NOW H,L=NUMBER OF BLOCKS
037C 113004             LXI     D,NUMBAD
037F CD8303             CALL    DCNV
0382 C9                 RET
                ;
0383 0620       DCNV    MVI     B,' '   ;SET FOR PLUS
0385 7C                 MOV     A,H
0386 B7                 ORA     A
0387 F29703             JP      H3
038A 062D               MVI     B,'-'
038C 7D                 MOV     A,L
038D 2F                 CMA
038E 3C                 INR     A
038F 6F                 MOV     L,A
0390 7C                 MOV     A,H
0391 2F                 CMA
0392 C29603             JNZ     H2
0395 3C                 INR     A
0396 67         H2      MOV     H,A
0397 22F703     H3      SHLD    DCNVHL
039A 3E20               MVI     A,' '
039C 12                 STAX    D
039D 78                 MOV     A,B
039E 32FB03             STA     DCNVPM
03A1 EB                 XCHG
03A2 22F903             SHLD    DCNVAD
03A5 AF                 XRA     A
03A6 32F603             STA     DCNVFL
03A9 01F0D8             LXI     B,-10000
03AC CDFC03             CALL    DFL8
03AF CDD303             CALL    DSTC
03B2 0118FC             LXI     B,-1000
03B5 CDFC03             CALL    DFL8
03B8 CDD303             CALL    DSTC
03BB 019CFF             LXI     B,-100
03BE CDFC03             CALL    DFL8
03C1 CDD303             CALL    DSTC
03C4 01F6FF             LXI     B,-10
03C7 CDFC03             CALL    DFL8
03CA CDD303             CALL    DSTC
03CD 3AF703             LDA     DCNVHL
```

```
                ;
                ; LOGICAL TO PHYSICAL MAPPING VECTOR
02AE 01070D1319LPMAP    DB      01,07,13,19,25,05,11,17,23,03,09,15,21
02BB 02080E141A         DB      02,08,14,20,26,06,12,18,24,04,10,16,22
                ;
                ; PUT THIS BLOCK IN THE BAD BLOCK LIST
02C8 2A4B04     SETBD   LHLD    DMCNT   ;GET NUM SECTORS
02CB 110800             LXI     D,BLOCK ;NUM IN THIS BLOCK
02CE 19                 DAD     D       ;BUMP BY NUM IN THIS BLOCK
02CF 224B04             SHLD    DMCNT   ;PUT NEW NUM SECTORS
02D2 2A4D04             LHLD    DMPTR   ;GET POINTER INTO DM
02D5 70                 MOV     M,B     ;PUT THIS BLOCK NUMBER IN DM
02D6 23                 INX     H       ;BUMP TO NEXT AVAIL EXTENT
02D7 224D04             SHLD    DMPTR   ;SAVE FOR NEXT TIME
02DA C9                 RET
                ;
                ; ELIMINATE ANY PREVIOUS [UNUSED],BAD ENTRIES
02DB 110C04     OPENB   LXI     D,BFCB  ;POINT TO BAD FCB
02DE 0E13               MVI     C,19    ;DELETE FILE
02E0 CD0500             CALL    BDOS
02E3 110C04             LXI     D,BFCB
02E6 0E16               MVI     C,22    ;MAKE FILE
02E8 CD0500             CALL    BDOS
02EB 110C04             LXI     D,BFCB
02EE 0E0F               MVI     C,15    ;OPEN FILE
02F0 CD0500             CALL    BDOS
02F3 FEFF               CPI     255     ;CHECK FOR ERROR
02F5 C0                 RNZ
02F6 11FC02             LXI     D,ERMSG3 ;SAY CANT OPEN
02F9 C31B01             JMP     PMSG
                ;
02FC 0D0A43414EERMSG3   DB      0DH,0AH,'CANT CREATE [UNUSED].
                                                        BAD',0DH,0AH,'$'
                ;
                ;  MOVE BAD AREA DM EXTENTS TO BFCB
0319 214F04     SETDM   LXI     H,DM
031C 224D04             SHLD    DMPTR
031F 2A4B04             LHLD    DMCNT
0322 7C         SETDM0  MOV     A,H
0323 B7                 ORA     A
0324 C22D03             JNZ     GOBIG
0327 7D                 MOV     A,L
0328 FE81               CPI     129
032A DA4F03             JC      SETDME
032D 1180FF     GOBIG   LXI     D,-128
0330 19                 DAD     D
0331 E5                 PUSH    H
0332 3E80               MVI     A,128
0334 CD4F03             CALL    SETDME
0337 EB                 XCHG
0338 224D04             SHLD    DMPTR
033B CD6403             CALL    CLOSEB
033E 3A4A04             LDA     FNUM
0341 3C                 INR     A
0342 324A04             STA     FNUM
0345 321404             STA     BFCB+8
0348 CDDB02             CALL    OPENB
034B E1                 POP     H
034C C32203             JMP     SETDM0
034F 2A4D04     SETDME  LHLD    DMPTR
0352 EB                 XCHG
0353 211C04             LXI     H,BFCB+16
0356 0610               MVI     B,16
0358 321B04             STA     BFCB+15 ;PUT RC IN PLACE
03D0 F630               ORI     '0'
03D2 5F                 MOV     E,A
03D3 2AF903     DSTC    LHLD    DCNVAD
03D6 3AF603             LDA     DCNVFL
03D9 B7                 ORA     A
03DA C2ED03             JNZ     DSTC3
03DD 83         DSTC1   ADD     E
03DE 32F603             STA     DCNVFL
03E1 C2E903             JNZ     DSTC2
03E4 3E20               MVI     A,' '
03E6 C3F003             JMP     DSTC4
03E9 3AFB03     DSTC2   LDA     DCNVPM
03EC 77                 MOV     M,A
03ED 3E30       DSTC3   MVI     A,'0'
03EF B3                 ORA     E
03F0 23         DSTC4   INX     H
03F1 77                 MOV     M,A
03F2 22F903             SHLD    DCNVAD
03F5 C9                 RET
03F6 00         DCNVFL  DB      0
03F7 0000       DCNVHL  DW      0
03F9 0000       DCNVAD  DW      0
03FB 00         DCNVPM  DB      0
03FC 2AF703     DFL8    LHLD    DCNVHL
03FF 1E00               MVI     E,0
0401 09         DF1     DAD     B
0402 7C                 MOV     A,H
0403 B7                 ORA     A
0404 F8                 RM
0405 1C                 INR     E
0406 22F703             SHLD    DCNVHL
0409 C30104             JMP     DF1
                ;
                ;
040C 005B554E55BFCB     DB      0,'[UNUSED]BAD',0,0,0,0
041C                    DS      17
                ;
042D 0D0A09     ENDMSG  DB      0DH,0AH,'       '
0430 202020204ENUMBAD   DB      '    NO'
0436 2042414420         DB      ' BAD BLOCKS FOUND',0DH,0AH,'$'
                ;
044A 30         FNUM    DB      '0'     ;USED IF MORE THAN 16 BAD BLOCKS
044B 0000       DMCNT   DW      0       ;NUMBER OF SECTORS IN TROUBLE
044D 4F04       DMPTR   DW      DM      ;POINT TO WHERE NEXT BLOCK ID GOES
                ;       ALLOCATION MAP FOR BAD SPOTS
044F 0000000000DM       DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
045F 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
046F 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
047F 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
048F 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
049F 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
04AF 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
04BF 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
04CF 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
04DF 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
04EF 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
04FF 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
050F 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
051F 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
052F 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
053F 0000000000         DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

054F                    END     100H
```

TEXT EDITOR (Continued from Page 95)

SAMPLE RUN

```
.J 0100
EDITOR V 2
BUFFER SIZE? 5
>BOTTOM
04FE 04FE 09FE
>TOP
>PRINT

END OF TEXT REACHED
>NEXT

END OF BUFFER REACHED
>INSERT
LOWER LIMIT TEST BACKSPACE THROUGH LEFT MARGIN
>I
UPPER LIMIT TEST GO THROUGH RIGHT MARGIN12345678901234567890123456789012345
END OF BUFFER REACHED
>T

>P 10

END OF TEXT REACHED
>INSERT
this is line 1
this is lline 2
this is lline three
this is line 4
this is line 5
this is the last line

>top

>print 20
this is line 1
this is lline 2
this is lline three
this is line 4
this is line 5
this is the last line

END OF TEXT REACHED
>c /ll/l/
this is line 2

>R
this is line three

>R

END OF TEXT REACHED
>t
>p 20
this is line 1
this is line 2
this is line three
this is line 4
this is line 5
this is the last line

END OF TEXT REACHED
>bottom
04FE 0563 09FE
>next -1
this is the last line
this is the last line
this is the last line
this is the last line
this is the last line

END OF TEXT REACHED
>p 256
ILLEGAL COMMAND
>c /the/not the/
this is not the last line

>R
this is not the last line

>R
this is not the last line

>R
this is not the last line

>R
this is not the last line

>R
this is not the last line

>R
this is not the last line

>R

END OF TEXT REACHED
>NEXT -255

END OF BUFFER REACHED
>print 255

this is the first line
this is line 1
this is line 2
this is line three
this is line 4
this is line 5
this is not the last line
this is not the last line
this is not the last line
this is not the last line
this is not the last line
this is not the last line
this is not the last line

END OF TEXT REACHED
>BOTTOM
04FE 061A 0BFE
>HERE

>B
04FE 0630 0BFE
>top

>print 255
this is the first line
this is line 1
this is line 2
this is line three
this is line 4
this is line 5
```

```
>move

>top

>here

>top

>p 20
this is the last line
this is line 1
this is line 2
this is line three
this is line 4
this is line 5
this is the last line

END OF TEXT REACHED
>c /last/first/
this is the first line

>b
04FE 057A 09FE
>h

>h

>h

>h

>h

>here

>t

>p 30
this is the first line
this is line 1
this is line 2
this is line three
this is line 4
this is line 5
this is the last line
this is the last line
this is the last line
this is the last line
this is the last line
this is the last line
this is the last line

END OF TEXT REACHED
>bottom
04FE 05FE 09FE
>quanity 2

>b
04FE 05FE 0BFE
>t

>find /5/
this is line 5

>n

>p 255
this is the last line
this is the last line
this is not the last line
this is not the last line
this is not the last line
this is not the last line
this is not the last line
this is not the last line
this is not the last line
this is the last line

END OF TEXT REACHED
>f /not the/
this is not the last line

>c /not //
this is the last line

>delete 5

>b
04FE 05AE 0BFE
>top

>print 255
this is the first line
this is line 1
this is line 2
this is line three
this is line 4
this is line 5
this is not the last line
this is not the last line
this is the last line

END OF TEXT REACHED
>n 6

>p

this is line 5

>p
this is not the last line

>d

>n -3

>p
this is line three

>c /three/3/
this is line 3

>top

>print 15
this is the first line
this is line 1
this is line 2
this is line 3
this is line 4
this is line 5
this is not the last line
this is the last line

END OF TEXT REACHED
>bottom
04FE 0590 0BFE
>x
```

```
.J 0103
>P 255

END OF TEXT REACHED
>T

>P 255
this is the first line
this is line 1
this is line 2
this is line 3
this is line 4
this is line 5
this is not the last line
this is the last line

END OF TEXT REACHED
>U
ILLEGAL COMMAND
>X
```

PROGRAM LISTING

```
00001                    NAM       EDITOR
00002                  * ROBERT HUDSON
00003                  * JANUARY 24, 1979
00004                    OPT       S,NOG,P
00005      FCD9        OUT4HS EQU  $FCD9
00006      F266        TVMASK EQU  $F266
00007      00F3        ECHO   EQU  $F3
00008      F000        ACIA   EQU  $F000
00009      0D0A        CRLF   EQU  $0D0A
00010      0004        EOT    EQU  4
00011                  * VARIABLE STORAGE AND WORK AREA
00012 0010             ORG       $10
00013 0010 0002        TOPBUF RMB  2         DO NOT REARRANGE
00014 0012 0002        CPNTR  RMB  2         USED TO PROVIDE BUFFER
00015 0014 0002        ENDBUF RMB  2         LIMITS WITH BOTTOM COMMAND
00016 0016 0002        TEMIDX RMB  2
00017 0018 0002        TEMBUF RMB  2
00018 001A 0002        STKSTR RMB  2
00019 001C 0001        LENGTH RMB  1
00020 001D 0001        NEGFLG RMB  1
00021 001E 0001        FLAG01 RMB  1
00022 001F 0001        DELIM  RMB  1
00023 0020 0002        TBUF1  RMB  2
00024 0022 0002        TBUF2  RMB  2
00025 0024 0001        MBUFL  RMB  1
00026 0025 0001        ADDCNT RMB  1
00027 0030             ORG       $30
00028                  * LINE BUFFER AREA
00029 0030 0048        LINBUF RMB  72
00030                  * MOVE BUFFER AREA
00031 0078 0048        MOVBUF RMB  72
00032                  * TEXT EDITOR PROGRAM
00033 0100             ORG       $0100
00034 0100 7E 01CA     SETUP  JMP  INIT      JUMP TABLE FOR
00035 0103 7E 01F8     HOT    JMP  START     SYSTEM LINKAGE
00036 0106 7E FCEB     INPUT  JMP  $FCEB
00037 0109 7E FCE3     OUTPUT JMP  $FCE3
00038 010C 7E F882     PDATA  JMP  $F882     ENTER2
00039                  * COMMAND TABLE        SYNTAX
00040 010F 50          CMDTBL FCB  'P        >P nnn
00041 0110 0258             FDB    PRINT
00042 0112 4E               FCB    'N        >N "-"nnn
00043 0113 02B3             FDB    NEXT
00112 01CF DF 12            STX    CPNTR
00113 01D1 CE 0143          LDX    #SIZMSG
00114 01D4 BD 010C          JSR    PDATA
00115 01D7 CE 0031          LDX    #LINBUF+1
00116 01DA 86 20            LDA A  #$20
00117 01DC A7 00            STA A  X
00118 01DE 08          SIZE1 INX
00119 01DF BD 0106          JSR    INPUT
00120 01E2 A7 00            STA A  X
00121 01E4 81 0D            CMP A  #$0D
00122 01E6 26 F6            BNE    SIZE1
00123 01E8 8D D2            BSR    SIZE
00124 01EA CE 0020          LDX    #TBUF1
00125 01ED 86 A0            LDA A  #160      CLEAR SOME VARIABLES PLUS
00126 01EF 6F 00       CLREND CLR  X         THE TEXT AND MOVE BUFFERS
00127 01F1 08               INX
00128 01F2 4A               DEC A
00129 01F3 26 FA            BNE    CLREND
00130 01F5 BD 02A5          JSR    ERASE
00131                  * RESTART
00132 01F8 8E A060     START  LDS    #$A060    ESTABLISH STACK AREA
00133 01FB CE 013F     BEGIN  LDX    #CMDHDR
00134 01FE BD 010C          JSR    PDATA
00135 0201 CE 0030          LDX    #LINBUF
00136 0204 BD 0106     GETCHR JSR    INPUT
00137 0207 81 5F            CMP A  #$5F      BACKSPACE?
00138 0209 26 08            BNE    CONT1
00139 020B 09               DEX
00140 020C 8C 002F          CPX    #LINBUF-1LOWER LIMIT ERROR
00141 020F 27 6A            BEQ    ERRCBL
00142 0211 20 F1            BRA    GETCHR
00143 0213 81 18       CONT1  CMP A  #$18      CANCEL INPUT LINE
00144 0215 27 E4            BEQ    BEGIN
00145                  * IMPLEMENTATION OF REPEAT FUNCTION.
00146                  * SYNTAX:  AFTER PROMPT CHARACTER PRESS "R"
00147                  * EXAMPLE:  >R
00148                  * WILL REPEAT PREVIOUS COMMAND
00149                  * FACILIATES A REPEAT FIND AND CHANGE COMMAND
00150 0217 81 52       REPEAT CMP A  #'R
00151 0219 26 10            BNE    CONT2
00152 021B 8C 0030          CPX    #LINBUF
00153 021E 26 0B            BNE    CONT2
00154 0220 8D 4E            BSR    CRFUNC
00155 0222 5F               CLR B                CLEAR NEGATIVE FLAG
00156 0223 D7 1D            STA B  NEGFLG        AND GO TO NEXT
00157 0225 5C               INC B                TEXT LINE
00158 0226 BD 02B5          JSR    NXTREP
00159 0229 20 0C            BRA    REPT
00160 022B A7 00       CONT2  STA A  X
00161 022D 08               INX
00162 022E 8C 0078          CPX    #MOVBUF
00163 0231 27 48            BEQ    ERRCBL    UPPER LIMIT ERROR
00164 0233 81 0D            CMP A  #$0D
00165 0235 26 CD            BNE    GETCHR
00166                  * BUFFER FILLED
00167 0237 CE 010F     REPT   LDX    #CMDTBL
00168 023A 86 5F            LDA A  #$5F      MASK FOR LOWER CASE
00169 023C 94 30            AND A  LINBUF    ALLOWS COMMANDS TO BE
00170 023E A1 00       COMPAR CMP A  X         UPPER OR LOWER CASE
00171 0240 27 10            BEQ    GDCMD
00172 0242 08               INX
00173 0243 08               INX
00174 0244 08               INX
00175 0245 8C 013F          CPX    #SPARE+3
00176 0248 26 F4            BNE    COMPAR
00177                  * ERROR
00178 024A CE 0161     ERROR  LDX    #CMDERR
00179 024D BD 010C     ERROUT JSR    PDATA     PRINT ERROR MESSAGE
```

```
00044 0115 41            FCB       'A        >A
00045 0116 04AE          FDB       APPEND
00046 0118 46            FCB       'F        >F /string/
00047 0119 03E3          FDB       FIND
00048 011B 43            FCB       'C        >C /string1/string2/
00049 011C 0465          FDB       CHANGE
00050 011E 49            FCB       'I        >I
00051 011F 036D          FDB       INSERT
00052 0121 44            FCB       'D        >D nnn
00053 0122 0347          FDB       DELETE
00054 0124 54            FCB       'T        >T
00055 0125 02AE          FDB       TOP
00056 0127 45            FCB       'E        >E
00057 0128 02A5          FDB       ERASE
00058 012A 42            FCB       'B        >B
00059 012B 028E          FDB       BOTTOM
00060 012D 53            FCB       'S        >S
00061 012E 04CE          FDB       SAVE
00062 0130 4C            FCB       'L        >L
00063 0131 04B1          FDB       LOAD
00064 0133 51            FCB       'Q        >Q nnn
00065 0134 01AE          FDB       BUFSIZ
00066 0136 4D            FCB       'M        >M
00067 0137 0482          FDB       MOVE
00068 0139 48            FCB       'H        >H
00069 013A 049C          FDB       HERE
00070 013C 00      SPARE FCB       0         "X" JUMP TO MONITOR
00071 013D 0000          FDB       0         ADDRESS OF MONITOR
00072              *                            ENTRY POINT.
00073              * "R" REPEAT
00074              *
00075 013F 0D0A    CMDHDR FDB      CRLF
00076              * PROMPT CHARACTER
00077 0141 3E            FCB       '>
00078 0142 04            FCB       EOT
00079              * SYSTEM MESSAGES
00080 0143 0D0A    SIZMSG FDB      CRLF
00081 0145 45            FCC       /EDITOR V 2  /
00082 0151 0D0A          FDB       CRLF
00083 0153 42            FCC       /BUFFER SIZE? /
00084 0160 04            FCB       EOT
00085 0161 49      CMDERR FCC      /ILLEGAL COMMAND/
00086 0170 04            FCB       EOT
00087 0171 0D0A    TXTMSG FDB      CRLF
00088 0173 45            FCC       /END OF TEXT REACHED/
00089 0186 04            FCB       EOT
00090 0187 0D0A    ENDMSG FDB      CRLF
00091 0189 45            FCC       /END OF BUFFER REACHED/
00092 019E 04            FCB       EOT
00093 019F 53      SYNMSG FCC      /SYNTAX ERROR/
00094 01AB 0D0A          FDB       CRLF
00095 01AD 04            FCB       EOT
00096              * PROGRAM BEGINS
00097 01AE DE 14   BUFSIZ LDX      ENDBUF    INCREASES TEXT BUFFER
00098 01B0 DF 18         STX       TEMBUF    AREA AND CLEARS
00099 01B2 BD 02EB       JSR       GETNUM    NEW AREA
00100 01B5 8D 0C         BSR       INCSIZ
00101 01B7 DE 18         LDX       TEMBUF
00102 01B9 7E 02A7       JMP       CLR01
00103 01BC BD 02EB SIZE  JSR       GETNUM
00104 01BF DE 10         LDX       TOPBUF
00105 01C1 DF 14         STX       ENDBUF
00106 01C3 7C 0014 INCSIZ INC      ENDBUF
00107 01C6 5A            DEC B
00108 01C7 26 FA         BNE       INCSIZ
00109 01C9 39            RTS
00110 01CA CE 04FE INIT  LDX       #PGEND+1
00111 01CD DF 10         STX       TOPBUF
00180 0250 20 A6         BRA       START
00181              * VALID CMD
00182 0252 EE 01   GDCMD LDX       1,X
00183 0254 AD 00         JSR       X
00184 0256 20 A3         BRA       BEGIN
00185              * PRINT N LINES
00186 0258 BD 02EB PRINT JSR       GETNUM
00187 025B DE 12         LDX       CPNTR
00188 025D A6 00   CHARLD LDA A    X
00189 025F 27 2A         BEQ       PNTOER
00190 0261 08            INX
00191 0262 81 0D         CMP A     #$0D
00192 0264 27 04         BEQ       DECLIN
00193 0266 8D 15         BSR       CHOUT
00194 0268 20 F3         BRA       CHARLD
00195 026A 8D 04   DECLIN BSR      CRFUNC
00196 026C 5A            DEC B
00197 026D 26 EE         BNE       CHARLD
00198 026F 39            RTS
00199 0270 36      CRFUNC PSH A
00200 0271 86 0D         LDA A     #$0D
00201 0273 8D 08         BSR       CHOUT
00202 0275 86 0A         LDA A     #$0A
00203 0277 8D 04         BSR       CHOUT
00204 0279 32            PUL A
00205 027A 39            RTS
00206 027B 20 68   ERRCBL BRA      OVRRUN    COMMAND BUFFER LENGTH ERR
00207              * CHARACTER OUTPUT AND BREAK ROUTINE
00208 027D 36      CHOUT PSH A
00209 027E B6 F000       LDA A     ACIA      BREAK ON ANY CHARACTER
00210 0281 47            ASR A               CONTROL PORT ADDRESS
00211 0282 32            PUL A
00212 0283 25 03         BCS       INTERP
00213 0285 7E 0109       JMP       OUTPUT
00214 0288 7E 01F8 INTERP JMP      START
00215              * POINTERS
00216 028B 7E 0356 PNTOER JMP      ET01
00217              * BOTTOM
00218 028E DE 12   BOTTOM LDX      CPNTR
00219 0290 A6 00   BLOAD LDA A     X
00220 0292 27 03         BEQ       BOT1
00221 0294 08            INX
00222 0295 20 F9         BRA       BLOAD
00223 0297 DF 12   BOT1  STX       CPNTR
00224 0299 CE 0010       LDX       #TOPBUF
00225 029C C6 03         LDA B     #3
00226 029E BD FCD9 STORAG JSR      OUT4HS    OUTPUT THE CURRENT
00227 02A1 5A            DEC B               LIMITS OF THE TEXT BUFFER
00228 02A2 26 FA         BNE       STORAG    AS THREE HEX NUMBERS
00229 02A4 39      CHRET RTS
00230              * ERASE
00231 02A5 DE 10   ERASE LDX       TOPBUF
00232 02A7 6F 00   CLR01 CLR       X
00233 02A9 08            INX
00234 02AA 9C 14         CPX       ENDBUF
00235 02AC 26 F9         BNE       CLR01
00236              * TOP
00237 02AE DE 10   TOP   LDX       TOPBUF
00238 02B0 DF 12         STX       CPNTR
00239 02B2 39            RTS
00240              * NEXT
00241 02B3 8D 36   NEXT  BSR       GETNUM    NEXT ROUTINE NOW WILL
00242 02B5 DE 12   NXTREP LDX      CPNTR     GO TO BOTH LIMITS OF
00243 02B7 86 0D         LDA A     #$0D      TEXT BUFFER EVEN IF NEXT
00244 02B9 7D 001D       TST       NEGFLG    ARGUMENT IS GREATER
00245 02BC 27 12         BEQ       LOOP3     THAN THE NUMBER OF
00246 02BE 9C 10   LOOP2 CPX       TOPBUF    LINES REQUIRED.
00247 02C0 27 23         BEQ       OVRRUN    ALSO "CPNTR" IS STORED
```

```
00248 02C2 09           DEX                 EACH TIME A CARRIER
00249 02C3 09     LOOP0 DEX                 RETURN IS PASSED.
00250 02C4 A1 00        CMP A     X
00251 02C6 26 FB        BNE       LOOP0
00252 02C8 08           INX
00253 02C9 DF 12        STX       CPNTR
00254 02CB 5A           DEC B
00255 02CC 26 F0        BNE       LOOP2
00256 02CE 39           RTS
00257 02CF 08     LOOP1 INX
00258 02D0 9C 14  LOOP3 CPX       ENDBUF
00259 02D2 27 11        BEQ       OVRRUN
00260 02D4 A1 00        CMP A     X
00261 02D6 26 F7        BNE       LOOP1
00262 02D8 08           INX
00263 02D9 DF 12        STX       CPNTR
00264 02DB 5A           DEC B
00265 02DC 26 F2        BNE       LOOP3
00266 02DE 39     LCRTS RTS
00267 02DF 27 04  LENCK BEQ       OVRRUN    CHECKS LINE LIMITS
00268 02E1 C1 48        CMP B     #72
00269 02E3 23 F9        BLS       LCRTS
00270 02E5 CE 0187 OVRRUN LDX     #ENDMSG
00271 02E8 7E 024D      JMP       ERROUT
00272             *
00273             * GETNUM HAS BEEN ENHANCED TO
00274             * EVALUATE A MAXIMUM NUMBER OF 255.
00275             * THE ORIGINAL PROGRAM ONLY PROVIDED
00276             * EVALUATION OF NUMBER TO 99.
00277             *
00278 02EB CE 0030 GETNUM LDX     #LINBUF
00279 02EE 5F           CLR B
00280 02EF D7 1D        STA B     NEGFLG
00281 02F1 08     NUM1  INX
00282 02F2 A6 00        LDA A     X
00283 02F4 81 0D        CMP A     #$0D
00284 02F6 27 33        BEQ       OUTONE
00285 02F8 81 20        CMP A     #$20
00286 02FA 26 F5        BNE       NUM1
00287 02FC 08           INX
00288 02FD A6 00        LDA A     X
00289 02FF 81 2D        CMP A     #'-
00290 0301 26 0A        BNE       NUM4
00291 0303 7C 001D      INC       NEGFLG
00292 0306 08     GETNXT INX
00293 0307 A6 00  NUM2  LDA A     X
00294 0309 81 0D  NUM3  CMP A     #$0D
00295 030B 27 1F        BEQ       NUMRTS
00296 030D 80 30  NUM4  SUB A     #$30
00297 030F 2B 1C        BMI       GT0ERR
00298 0311 81 09        CMP A     #9
00299 0313 2E 18        BGT       GT0ERR
00300 0315 36     MUL10 PSH A               BASICALLY A MULTIPLY
00301 0316 86 09        LDA A     #9        BY 10 ROUTINE
00302 0318 97 25        STA A     ADDCNT
00303 031A 17           TBA
00304 031B 0C           CLC
00305 031C 1B     ADDBA ABA
00306 031D 25 0E        BCS       GT0ERR
00307 031F 7A 0025      DEC       ADDCNT
00308 0322 26 F8        BNE       ADDBA
00309 0324 33           PUL B
00310 0325 1B           ABA
00311 0326 25 05        BCS       GT0ERR
00312 0328 16           TAB
00313 0329 20 DB        BRA       GETNXT
00314 032B 5C     OUTONE INC B
00315 032C 39     NUMRTS RTS
```

```
00384 03B1 D0 1C        SUB B     LENGTH
00385 03B3 24 01        BCC       MOV02
00386 03B5 4A           DEC A
00387 03B6 97 16  MOV02 STA A     TEMIDX
00388 03B8 D7 17        STA B     TEMIDX+1
00389 03BA DE 16        LDX       TEMIDX
00390 03BC 08           INX
00391 03BD D6 1C        LDA B     LENGTH
00392 03BF F7 03C6      STA B     MOV04+1
00393 03C2 09     MOV03 DEX
00394 03C3 A6 00        LDA A     X
00395 03C5 A7 00  MOV04 STA A     X
00396 03C7 9C 12        CPX       CPNTR
00397 03C9 26 F7        BNE       MOV03
00398 03CB 39           RTS
00399             * LINE TO TEXT BUFFER
00400 03CC 9F 1A  PUTBUF STS      STKSTR
00401 03CE 9E 18        LDS       TEMBUF
00402 03D0 34           DES
00403 03D1 D6 1C        LDA B     LENGTH
00404 03D3 27 0B        BEQ       PUTRET
00405 03D5 DE 12        LDX       CPNTR
00406 03D7 32     PULL1 PUL A
00407 03D8 A7 00        STA A     X
00408 03DA 08           INX
00409 03DB 5A           DEC B
00410 03DC 26 F9        BNE       PULL1
00411 03DE DF 12        STX       CPNTR
00412 03E0 9E 1A  PUTRET LDS      STKSTR
00413 03E2 39           RTS
00414             * FIND
00415 03E3 CE 0030 FIND LDX       #LINBUF
00416 03E6 DF 20        STX       TBUF1
00417 03E8 8D 47        BSR       DEFBUF
00418 03EA 8D 11        BSR       MATCH
00419 03EC DE 12  BKMOV LDX       CPNTR
00420 03EE 86 0D        LDA A     #$0D
00421 03F0 09     DECR  DEX
00422 03F1 A1 00        CMP A     X
00423 03F3 26 FB        BNE       DECR
00424 03F5 08           INX
00425 03F6 DF 12        STX       CPNTR
00426 03F8 C6 01        LDA B     #1
00427 03FA 7E 025D      JMP       CHARLD
00428             * MATCH
00429 03FD 9F 1A  MATCH STS       STKSTR
00430 03FF DE 22        LDX       TBUF2
00431 0401 08           INX
00432 0402 9E 12        LDS       CPNTR
00433 0404 9F 1D        STS       NEGFLG
00434             *
00435 0406 34           DES                 ADJUST STACK
00436             *
00437 0407 D6 1C  GETLEN LDA B    LENGTH
00438 0409 9F 16        STS       TEMIDX
00439 040B 32     PULLCH PUL A
00440 040C 4D           TST A
00441 040D 27 19        BEQ       ENDSCH
00442 040F A1 00        CMP A     X
00443 0411 27 08        BEQ       FNDONE
00444 0413 9E 16        LDS       TEMIDX
00445 0415 DE 22        LDX       TBUF2
00446 0417 31           INS
00447 0418 08           INX
00448 0419 20 EC        BRA       GETLEN
00449 041B D1 1C  FNDONE CMP B    LENGTH
00450 041D 26 02        BNE       FNDTWO
00451 041F 9F 12        STS       CPNTR
00452 0421 08     FNDTWO INX
```

```
00316 032D 7E 024A GTOERR JMP       ERROR
00317              * DELETE
00318 0330 DE 12   DLINE  LDX       CPNTR     SUBROUTINE CREATED
00319 0332 7F 001C        CLR       LENGTH    TO ASSIST MOVE COMMAND
00320 0335 A6 00   GETONE LDA A     X
00321 0337 27 0D          BEQ       DELRTS
00322 0339 81 0D          CMP A     #$0D
00323 033B 27 06          BEQ       LINEND
00324 033D 7C 001C        INC       LENGTH
00325 0340 08             INX
00326 0341 20 F2          BRA       GETONE
00327 0343 7C 001C LINEND INC       LENGTH
00328 0346 39      DELRTS RTS
00329 0347 8D A2   DELETE BSR       GETNUM
00330 0349 8D E5   DLINE1 BSR       DLINE
00331 034B 4D             TST A
00332 034C 27 06          BEQ       ENDTXT
00333 034E 8D 0C          BSR       DLTLIN
00334 0350 5A             DEC B
00335 0351 26 F6          BNE       DLINE1
00336 0353 39             RTS
00337 0354 8D 06   ENDTXT BSR       DLTLIN
00338 0356 CE 0171 ET01   LDX       #TXTMSG
00339 0359 7E 024D        JMP       ERROUT
00340 035C DE 12   DLTLIN LDX       CPNTR
00341 035E 96 1C          LDA A     LENGTH
00342 0360 B7 0364        STA A     DLET+1
00343 0363 A6 00   DLET   LDA A     X
00344 0365 A7 00          STA A     X
00345 0367 08             INX
00346 0368 9C 14          CPX       ENDBUF
00347 036A 26 F7          BNE       DLET
00348 036C 39             RTS
00349              * INSERT
00350 036D CE 0030 INSERT LDX       #LINBUF
00351 0370 5F             CLR B
00352 0371 D7 1E          STA B     FLAG01
00353 0373 BD 0106 INS01  JSR       INPUT
00354 0376 81 1B          CMP A     #$1B      ESC  END OF INPUT?
00355 0378 26 05          BNE       INS02
00356 037A 7C 001E        INC       FLAG01
00357 037D 20 1E          BRA       INS05
00358 037F 81 18   INS02  CMP A     #$18      CANCEL CURRENT LINE?
00359 0381 26 05          BNE       INS03
00360 0383 BD 0270        JSR       CRFUNC
00361 0386 20 E5          BRA       INSERT
00362 0388 81 5F   INS03  CMP A     #$5F      BACKSPACE?
00363 038A 26 06          BNE       INS04
00364 038C 5A             DEC B
00365 038D 2B 1D          BMI       INRTS     LOWER LIMIT ERROR
00366 038F 09             DEX
00367 0390 20 E1          BRA       INS01
00368 0392 5C      INS04  INC B
00369 0393 BD 02DF        JSR       LENCK     HAS UPPER LIMIT OF
00370 0396 A7 00          STA A     X         INPUT BUFFER BEEN
00371 0398 08             INX                 EXCEEDED?
00372 0399 81 0D          CMP A     #$0D
00373 039B 26 D6          BNE       INS01
00374 039D D7 1C   INS05  STA B     LENGTH
00375 039F 8D 0C          BSR       MOVLIN
00376 03A1 CE 0030        LDX       #LINBUF
00377 03A4 DF 18          STX       TEMBUF
00378 03A6 8D 24          BSR       PUTBUF
00379 03A8 96 1E          LDA A     FLAG01
00380 03AA 27 C1          BEQ       INSERT
00381 03AC 39      INRTS  RTS
00382 03AD 96 14   MOVLIN LDA A     ENDBUF
00383 03AF D6 15          LDA B     ENDBUF+1
00453 0422 5A             DEC B
00454 0423 26 E6          BNE       PULLCH
00455 0425 9E 1A   FOUND  LDS       STKSTR
00456 0427 39             RTS
00457 0428 DE 1D   ENDSCH LDX       NEGFLG
00458 042A DF 12          STX       CPNTR
00459 042C 9E 1A          LDS       STKSTR
00460 042E 7E 0356        JMP       ET01
00461              * STRING
00462 0431 DE 20   DEFBUF LDX       TBUF1
00463 0433 08      INC03  INX
00464 0434 A6 00          LDA A     X
00465 0436 81 20          CMP A     #$20
00466 0438 27 06          BEQ       SPACE1
00467 043A 81 0D          CMP A     #$0D
00468 043C 27 21          BEQ       SYNERR
00469 043E 20 F3          BRA       INC03
00470 0440 08      SPACE1 INX
00471 0441 A6 00          LDA A     X
00472 0443 81 20          CMP A     #$20
00473 0445 27 F9          BEQ       SPACE1
00474 0447 97 1F          STA A     DELIM
00475 0449 DF 22   C0ENT  STX       TBUF2
00476 044B 5F             CLR B
00477 044C 08      INC04  INX
00478 044D A6 00          LDA A     X
00479 044F 91 1F          CMP A     DELIM
00480 0451 27 07          BEQ       ENDSTR
00481 0453 81 0D          CMP A     #$0D
00482 0455 27 08          BEQ       SYNERR
00483 0457 5C             INC B
00484 0458 20 F2          BRA       INC04
00485 045A DF 20   ENDSTR STX       TBUF1
00486 045C D7 1C          STA B     LENGTH
00487 045E 39             RTS
00488 045F CE 019F SYNERR LDX       #SYNMSG
00489 0462 7E 024D        JMP       ERROUT
00490              * CHANGE
00491 0465 CE 0030 CHANGE LDX       #LINBUF
00492 0468 DF 20          STX       TBUF1
00493 046A 8D C5          BSR       DEFBUF
00494 046C 8D 8F          BSR       MATCH
00495 046E BD 035C        JSR       DLTLIN
00496 0471 DE 20          LDX       TBUF1
00497 0473 8D D4          BSR       C0ENT
00498 0475 BD 03AD        JSR       MOVLIN
00499 0478 DE 22          LDX       TBUF2
00500 047A 08             INX
00501 047B DF 18          STX       TEMBUF
00502 047D 8D 2C          BSR       JPBUF
00503 047F 7E 03EC        JMP       BKMOV
00504              * MOVE: WILL STORE IN MOVE BUFFER THE LINE POINTED
00505              * TO BY "CPNTR" - CURRENT LINE POINTER.
00506              * ORIGINAL LINE CAN THEN BE DELETED BY "D" COMMAND.
00507              * OR BY NOT DELETING "MOVE" CAN BE USED
00508              * TO DUPLICATE LINES.
00509              * SYNTAX:  >MOVE
00510              * USED WITH THE HERE COMMAND.
00511 0482 DE 12   MOVE   LDX       CPNTR     FIND LINE LENGTH,
00512 0484 DF 18          STX       TEMBUF    IF > 72 THEN DO NOT MOVE
00513 0486 BD 0330        JSR       DLINE
00514 0489 D6 1C          LDA B     LENGTH
00515 048B BD 02DF        JSR       LENCK
00516 048E D7 24          STA B     MBUFL
00517 0490 CE 0078        LDX       #MOVBUF
00518 0493 DF 12          STX       CPNTR
00519 0495 8D 14          BSR       JPBUF
00520 0497 DE 18          LDX       TEMBUF
00521 0499 DF 12          STX       CPNTR
```

```
00522 049B 39              RTS
00523                    * HERE: COMMAND USED TO INSERT CURRENT LINE OF
00524                    * TEXT IN "MOVE" BUFFER TO LINE IN TEXT
00525                    * BUFFER POINTED TO BY "CPNTR".
00526                    * SYNTAX: AFTER "MOVE" COMMAND USE "TOP", "BOTTOM"
00527                    * OR "NEXT" COMMAND TO POSITION
00528                    * CURRENT LINE POINTER TO DESIRED LINE TO INSERT
00529                    * TEXT STORED IN "MOVE" BUFFER.
00530                    * >MOVE          :STORE LINE IN MOVE BUFFER
00531                    * >DELETE        :DELETE EXISTING LINE
00532                    * >NEXT -5       :GO BACK FIVE LINES
00533                    * >HERE          :INSERT LINE STORED IN "MOVE" BUFFER.
00534 049C D6 24   HERE   LDA B    MBUFL     IF ZERO THEN ERROR
00535 049E BD 02DF        JSR      LENCK
00536 04A1 D7 1C          STA B    LENGTH
00537 04A3 BD 03AD        JSR      MOVLIN
00538 04A6 CE 0078        LDX      #MOVBUF
00539 04A9 DF 18          STX      TEMBUF
00540 04AB 7E 03CC JPBUF  JMP      PUTBUF
00541                    * APPEND
00542 04AE BD 028E APPEND JSR      BOTTOM
00543                    * LOAD
00544 04B1 DE 12   LOAD   LDX      CPNTR
00545 04B3 86 FF          LDA A    #$FF
00546 04B5 97 F3          STA A    ECHO      TURN OFF INPUT ECHO
00547 04B7 BD 0106 LOAD1  JSR      INPUT
00548 04BA 4D             TST A              SKIP OVER NULLS
00549 04BB 27 FA          BEQ      LOAD1
00550 04BD 81 03          CMP A    #3        ARE WE FINISHED?
00551 04BF 27 07          BEQ      LEND
00552 04C1 A7 00          STA A    X
00553 04C3 08             INX
00554 04C4 9C 14          CPX      ENDBUF    CHECK TO BE SURE
00555 04C6 26 EF          BNE      LOAD1     END OF TEXT BUFFER IS
00556 04C8 7F 00F3 LEND   CLR      ECHO      NOT EXCEEDED. RESTORE
00557 04CB 7E 02AE        JMP      TOP       ECHO AND RETURN.
00558                    * SAVE
00559 04CE 7F F266 SAVE   CLR      TVMASK    TURN OFF CRT
00560 04D1 8D 20          BSR      NULLS     OUTPUT LEADER
00561 04D3 DE 12          LDX      CPNTR     BEGIN AT CURRENT
00562 04D5 09             DEX                LINE POINTER AND OUTPUT
00563 04D6 08      MORES  INX                DATA. CHECK BUFFER
00564 04D7 A6 00          LDA A    X         LIMITS AND END IF
00565 04D9 27 09          BEQ      ENDS      REACHED.
00566 04DB 9C 14          CPX      ENDBUF
00567 04DD 27 05          BEQ      ENDS
00568 04DF BD 0109        JSR      OUTPUT
00569 04E2 20 F2          BRA      MORES
00570 04E4 8D 0D   ENDS   BSR      NULLS     OUTPUT TRAILER
00571 04E6 86 03          LDA A    #3        OUTPUT END OF FILE
00572 04E8 BD 0109        JSR      OUTPUT
00573 04EB 86 80          LDA A    #$80
00574 04ED B7 F266        STA A    TVMASK    TURN CRT ON
00575 04F0 7E 02AE        JMP      TOP       AND RETURN.
00576 04F3 C6 32   NULLS  LDA B    #50
00577 04F5 4F             CLR A
00578 04F6 BD 0109 MNULLS JSR      OUTPUT
00579 04F9 5A             DEC B
00580 04FA 26 FA          BNE      MNULLS
00581 04FC 39             RTS
00582 04FD 0D      PGEND  FCB      $0D       TEXT BUFFER
00583                    END

OUT4HS FCD9
TVMASK F266
ECHO   00F3
ACIA   F000
CRLF   0D0A
EOT    0004
```

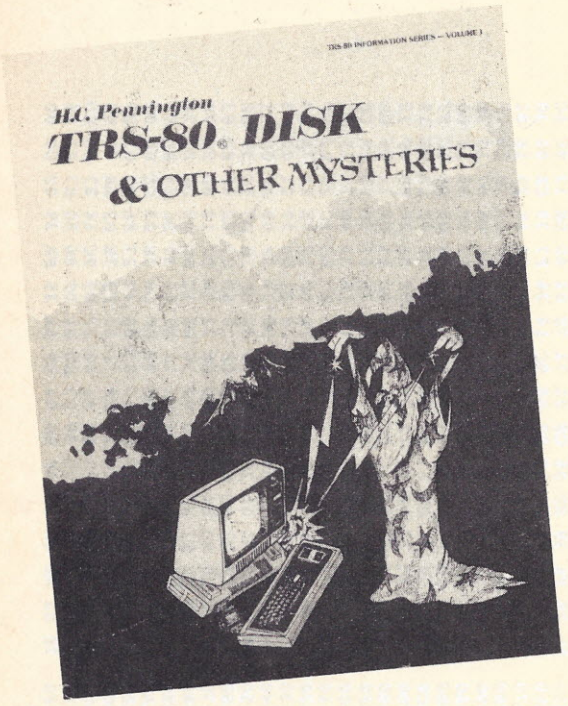
```
LCRTS  02DE
LENCK  02DF
OVRRUN 02E5
GETNUM 02EB
NUM1   02F1
GETNXT 0306
NUM2   0307
NUM3   0309
NUM4   030D
MUL10  0315
ADDBA  031C
OUTONE 032B
NUMRTS 032C
GT0ERR 032D
DLINE  0330
GETONE 0335
LINEND 0343
DELRTS 0346
DELETE 0347
DLINE1 0349
ENDTXT 0354
ET01   0356
DLTLIN 035C
DLET   0363
INSERT 036D
INS01  0373
INS02  037F
INS03  0388
INS04  0392
INS05  039D
INRTS  03AC
MOVLIN 03AD
MOV02  03B6
MOV03  03C2
MOV04  03C5
PUTBUF 03CC
PULL1  03D7
PUTRET 03E0
FIND   03E3
BKMOV  03EC
DECR   03F0
MATCH  03FD
GETLEN 0407
PULLCH 040B
FNDONE 041B
FNDTWO 0421
FOUND  0425
ENDSCH 0428
DEFBUF 0431
INC03  0433
SPACE1 0440
C0ENT  0449
INC04  044C
ENDSTR 045A
SYNERR 045F
CHANGE 0465
MOVE   0482
HERE   049C
JPBUF  04AB
APPEND 04AE
LOAD   04B1
LOAD1  04B7
LEND   04C8
SAVE   04CE
MORES  04D6
ENDS   04E4
NULLS  04F3
MNULLS 04F6
PGEND  04FD
```

```
TOPBUF 0010
CPNTR  0012
ENDBUF 0014
TEMIDX 0016
TEMBUF 0018
STKSTR 001A
LENGTH 001C
NEGFLG 001D
FLAG01 001E
DELIM  001F
TBUF1  0020
TBUF2  0022
MBUFL  0024
ADDCNT 0025
LINBUF 0030
MOVBUF 0078
SETUP  0100
HOT    0103
INPUT  0106
OUTPUT 0109
PDATA  010C
CMDTBL 010F
SPARE  013C
CMDHDR 013F
SIZMSG 0143
CMDERR 0161
TXTMSG 0171
ENDMSG 0187
SYNMSG 019F
BUFSIZ 01AE
SIZE   01BC
INCSIZ 01C3
INIT   01CA
SIZE1  01DE
CLREND 01EF
START  01F8
BEGIN  01FB
GETCHR 0204
CONT1  0213
REPEAT 0217
CONT2  022B
REPT   0237
COMPAR 023E
ERROR  024A
ERROUT 024D
GDCMD  0252
PRINT  0258
CHARLD 025D
DECLIN 026A
CRFUNC 0270
ERRCBL 027B
CHOUT  027D
INTERP 0288
PNTOER 028B
BOTTOM 028E
BLOAD  0290
BOT1   0297
STORAG 029E
CHRET  02A4
ERASE  02A5
CLR01  02A7
TOP    02AE
NEXT   02B3
NXTREP 02B5
LOOP2  02BE
LOOP0  02C3
LOOP1  02CF
LOOP3  02D0
```

```
.D  ?0100 ?04FF
0100  7E 01 CA 7E 01 F8 7E FC EB 7E FC E3 7E F8 82 50
0110  02 58 4E 02 B3 41 04 AE 46 03 E3 43 04 65 49 03
0120  6D 44 03 47 54 02 AE 45 02 A5 42 02 8E 53 04 CE
0130  4C 04 B1 51 01 AE 4D 04 82 48 04 9C 00 00 00 0D
0140  0A 3E 04 0D 0A 45 44 49 54 4F 52 20 56 20 32 20
0150  20 0D 0A 42 55 46 46 45 52 20 53 49 5A 45 3F 20
0160  04 49 4C 4C 45 47 41 4C 20 43 4F 4D 4D 41 4E 44
0170  04 0D 0A 45 4E 44 20 4F 46 20 54 45 58 54 20 52
0180  45 41 43 48 45 44 04 0D 0A 45 4E 44 20 4F 46 20
0190  42 55 46 46 45 52 20 52 45 41 43 48 45 44 04 53
01A0  59 4E 54 41 58 20 45 52 52 4F 52 0D 0A 04 DE 14
01B0  DF 18 BD 02 EB 8D 0C DE 18 7E 02 A7 BD 02 EB DE
01C0  10 DF 14 7C 00 14 5A 26 FA 39 CE 04 FE DF 10 DF
01D0  12 CE 01 43 BD 01 0C CE 00 31 86 20 A7 00 08 BD
01E0  01 06 A7 00 81 0D 26 F6 8D D2 CE 00 20 86 A0 6F
01F0  00 08 4A 26 FA BD 02 A5 8E A0 60 CE 01 3F BD 01
0200  0C CE 00 30 BD 01 06 81 5F 26 08 09 8C 00 2F 27
0210  6A 20 F1 81 18 27 E4 81 52 26 10 8C 00 30 26 0B
0220  8D 4E 5F D7 1D 5C BD 02 B5 20 0C A7 00 08 8C 00
0230  78 27 48 81 0D 26 CD CE 01 0F 86 5F 94 30 A1 00
0240  27 10 08 08 08 8C 01 3F 26 F4 CE 01 61 BD 01 0C
0250  20 A6 EE 01 AD 00 20 A3 BD 02 EB DE 12 A6 00 27
0260  2A 08 81 0D 27 04 8D 15 20 F3 8D 04 5A 26 EE 39
0270  36 86 0D 8D 08 86 0A 8D 04 32 39 20 68 36 B6 F0
0280  00 47 32 25 03 7E 01 09 7E 01 F8 7E 03 56 DE 12
0290  A6 00 27 03 08 20 F9 DF 12 CE 00 10 C6 03 BD FC
02A0  D9 5A 26 FA 39 DE 10 6F 00 08 9C 14 26 F9 DE 10
02B0  DF 12 39 8D 36 DE 12 86 0D 7D 00 1D 27 12 9C 10
02C0  27 23 09 09 A1 00 26 FB 08 DF 12 5A 26 F0 39 08
02D0  9C 14 27 11 A1 00 26 F7 08 DF 12 5A 26 F2 39 27
02E0  04 C1 48 23 F9 CE 01 87 7E 02 4D CE 00 30 5F D7
02F0  1D 08 A6 00 81 0D 27 33 81 20 26 F5 08 A6 00 81
0300  2D 26 0A 7C 00 1D 08 A6 00 81 0D 27 1F 80 30 2B
0310  1C 81 09 2E 18 36 86 09 97 25 17 0C 1B 25 0E 7A
0320  00 25 26 F8 33 1B 25 05 16 20 DB 5C 39 7E 02 4A
0330  DE 12 7F 00 1C A6 00 27 0D 81 0D 27 06 7C 00 1C
0340  08 20 F2 7C 00 1C 39 8D A2 8D E5 4D 27 06 8D 0C
0350  5A 26 F6 39 8D 06 CE 01 71 7E 02 4D DE 12 96 1C
0360  B7 03 64 A6 00 A7 00 08 9C 14 26 F7 39 CE 00 30
0370  5F D7 1E BD 01 06 81 1B 26 05 7C 00 1E 20 1E 81
0380  18 26 05 BD 02 70 20 E5 81 5F 26 06 5A 2B 1D 09
0390  20 E1 5C BD 02 DF A7 00 08 81 0D 26 D6 D7 1C 8D
03A0  0C CE 00 30 DF 18 8D 24 96 1E 27 C1 39 96 14 D6
03B0  15 D0 1C 24 01 4A 97 16 D7 17 DE 16 08 D6 1C F7
03C0  03 C6 09 A6 00 A7 00 9C 12 26 F7 39 9F 1A 9E 18
03D0  34 D6 1C 27 0B DE 12 32 A7 00 08 5A 26 F9 DF 12
03E0  9E 1A 39 CE 00 30 DF 20 8D 47 8D 11 DE 12 86 0D
03F0  09 A1 00 26 FB 08 DF 12 C6 01 7E 02 5D 9F 1A DE
0400  22 08 9E 12 9F 1D 34 D6 1C 9F 16 32 4D 27 19 A1
0410  00 27 08 9E 16 DE 22 31 08 20 EC D1 1C 26 02 9F
0420  12 08 5A 26 E6 9E 1A 39 DE 1D DF 12 9E 1A 7E 03
0430  56 DE 20 08 A6 00 81 20 27 06 81 0D 27 21 20 F3
0440  08 A6 00 81 20 27 F9 97 1F DF 22 5F 08 A6 00 91
0450  1F 27 07 81 0D 27 08 5C 20 F2 DF 20 D7 1C 39 CE
0460  01 9F 7E 02 4D CE 00 30 DF 20 8D C5 8D 8F BD 03
0470  5C DE 20 8D D4 BD 03 AD DE 22 08 DF 18 8D 2C 7E
0480  03 EC DE 12 DF 18 BD 03 30 D6 1C BD 02 DF D7 24
0490  CE 00 78 DF 12 8D 14 DE 18 DF 12 39 D6 24 BD 02
04A0  DF D7 1C BD 03 AD CE 00 78 DF 18 7E 03 CC BD 02
04B0  8E DE 12 86 FF 97 F3 BD 01 06 4D 27 FA 81 03 27
04C0  07 A7 00 08 9C 14 26 EF 7F 00 F3 7E 02 AE 7F F2
04D0  66 8D 20 DE 12 09 08 A6 00 27 09 9C 14 27 05 BD
04E0  01 09 20 F2 8D 0D 86 03 BD 01 09 86 80 B7 F2 66
04F0  7E 02 AE C6 32 4F BD 01 09 5A 26 FA 39 0D 00 00
 ?
```

THE **HOTTEST**

"How To Fix It . . ."
EVER WRITTEN!!



This valuable textbook, the first in a series by H. C. Pennington, has been reviewed in 3 major magazines and has been the topic of extensive conversation on two major computer communications networks. Presently, the book is being distributed and sold throughout the United States, Canada, the Union of South Africa, England, Germany, Australia, the Netherlands & others. William Barden, Jr., a well known and widely published computer author comments, ". . . my advice is to get it and USE it!"

Easy to read, entertaining material covers nearly every aspect of the many problems encountered on the TRS-80 system with disk storage. Actual examples, problem samples, memory maps, "how to fix . . ." solutions and countless illustrations are presented throughout.

Author: H. C. Pennington Publisher: IJG Computer Services
First Edition - Second Printing - Soft Cover - 130 Page
8½" x 11 — Wt. 1 lb. — Retail price: $22.50

FOR TRS-80 USERS DATA DYNAMICS TECHNOLOGY

Please rush me _____ copy/copies of **TRS-80 Disk & Other Mysteries** @ $22.50 each. Add $1.00 shipping & handling per book.

☐ AMERICAN EXPRESS ☐ VISA ☐ master charge ☐ Check

Exp. Date_____ Card # _____

Signature _____

Name (Print) _____

Address _____

City_____ State_____ Zip_____

TOTAL ORDER $_____

TAX $_____

SHIPPING & HANDLING $_____

TOTAL ENCLOSED $_____

DATA DYNAMICS TECHNOLOGY
P.O. Box 1217, Cerritos, CA 90701

For European delivery contact **INTERFACE AGE Europe,** Dahlienstr. 4, D-8011 Munchen-Vaterstetten, West Germany.
*California residents add 6% sales tax. Availability and prices quoted subject to change without notice.
Please allow six weeks for delivery. You may photocopy this page if you wish to keep your INTERFACE AGE intact.
Orders *cannot* be shipped unless accompanied by payment, including shipping & handling and tax where applicable.

DATA DYNAMICS TECHNOLOGY
A Division of INTERFACE AGE Magazine (213) 926-9548

9/80

TRS-80 DISK & OTHER MYSTERIES TABLE OF CONTENTS

Free Literature

- **Insulation Displacement Connectors.** An 18-page catalog, Mas/ter-IDC-2, includes charts on contact arrangement, standard characteristic data and ordering instructions. ITT Cannon Electric, 666 E. Dyer Rd., Santa Ana, CA 92702. CIRCLE INQUIRY NO. 209

- **Eprom Testing Program.** An 8-page brochure describes the Intel method for testing reliability of erasable programmable read-only memories. Intel, Literature Dept., 3065 Bowers Ave., Santa Clara, CA 95051. CIRCLE INQUIRY NO. 210

- **Pascal Language.** An 8-page pamphlet, 'Pascal, a Programming Language for Today,' details Ramtek's approach to Pascal. A comparison chart provides a feature-by-feature breakdown of languages including Basic, Fortran, and Cobol. Ramtek, 2211 Lawson Lane, Santa Clara, CA 95050. CIRCLE INQUIRY NO. 211

- **Microprocessor Course.** A 4-page folder describes thirty 30-minute color videotaped lectures on the programming of microprocessors. For design engineers or technicians who need to develop or maintain microprocessor-oriented systems. W.L. Somervell, Jr., ERG Director, Colorado State U., Christman Field, Bldg. 1000, Ft. Collins, CO 80253. CIRCLE INQUIRY NO. 212

- **Micro Problems.** A catalog describes microcomputer interference control products and protective devices. Typical applications and uses are also outlined. Request Catalog 801. Electronic Specialists, Inc., 171 S. Main St., Natick, MA 01760. CIRCLE INQUIRY NO. 213

- **Line Printer Systems.** A 2-page catalog sheet describes the DAC D-2200 series. Complete model and specification data are provided. D-2200 Bulletin, Digital Assoc. Corp., 1039 E. Main St., Stamford, CT 06902. CIRCLE INQUIRY NO. 214

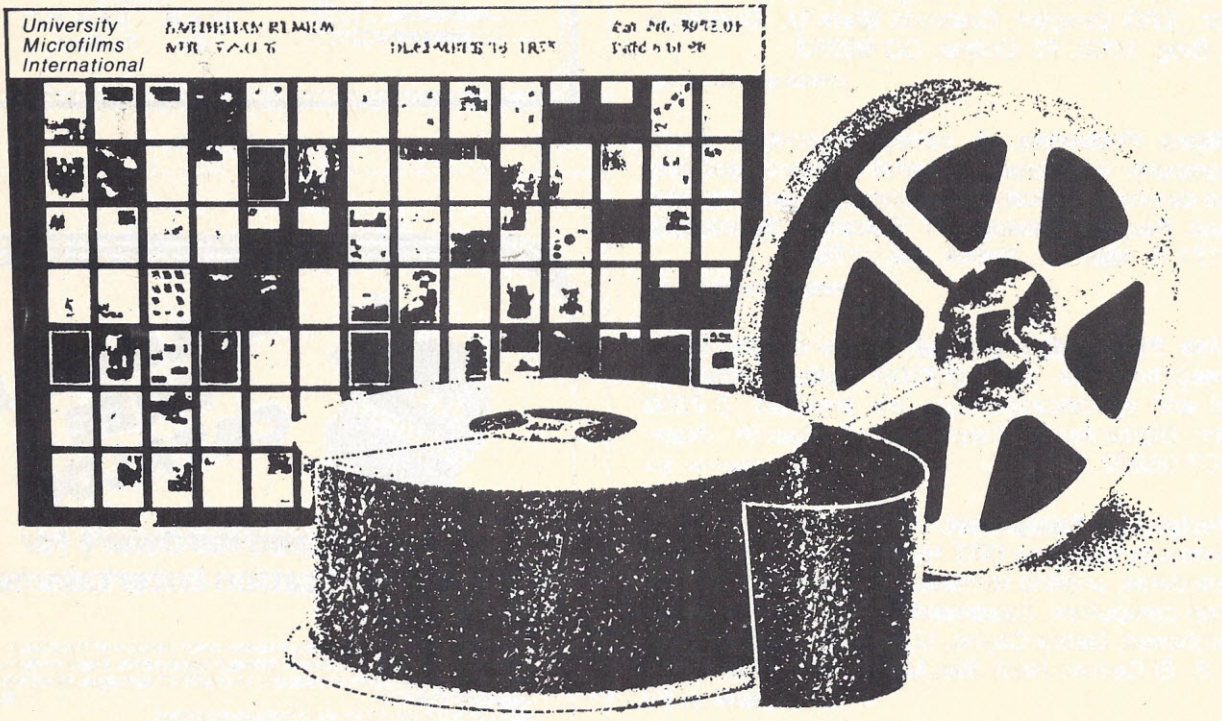
- **Peripheral Equipment.** A 12-page catalog lists products including CRT terminals, acoustic couplers, modems, printing terminals, storage devices, and desktop computers. Equipment can be rented, leased or purchased. Debra Coyne, U.S. Instrument Rentals, 2121 S. El Camino Real, San Mateo, CA 94403. CIRCLE INQUIRY NO. 215

- **Power Supplies and Converters.** An 8-page catalog provides electrical/mechanical specifications, features, photos, and ordering information for AC-DC linear and switching power supplies and DC-DC converters. Power Products Div., Computer Products, Inc., 1400 N.W. 70 St., Ft. Lauderdale, FL 33309. CIRCLE INQUIRY NO. 216

- **Continuous Form Set.** A 4-page brochure describes the Speedi-form multi-purpose multi-part continuous form set. Moore Business Forms, Inc., Corp. Communications, 1205 N. Milwaukee Ave., Glenview, IL 60025. CIRCLE INQUIRY NO. 217

INTERFACE AGE™

is available in microfom



Please send me additional information.

Name _____

Institution _____

Street _____

City _____

State _____ Zip _____

University Microfilms International

300 North Zeeb Road
Dept. P.R.
Ann Arbor, MI 48106
U.S.A.

18 Bedford Row
Dept. P.R.
London, WC1R 4EJ
England

MICRO MARKET

MICRODEX

MICRO INDEX CLASSIFIED

FOR SALE: Unused Polymorphic System 8813 with 32K RAM memory with floppy disk drives, printer interface and Abern-Sopher Multiwriter III. System has Canadian import tax paid. Offers for complete system to: Bishop Management, #8 - 825 McBride Blvd., New Westminster, B.C., Canada, V3L 5B5. (604) 525-8148.

PASCAL LONGINTEGERS for North Star. Arithmetics to 36 digits. Z80 routines install in seconds. Very fast. Pascal demonstration included. $20. VSS, 257 5th Ave., San Francisco, CA 94118, (415) 752-9775.

TRS-80: TEACHER developed software. 16K Level II, grades 6-9. "Largo Island"-stimulating population simulation, and "Metric Momma"—ISU the easy way. $10 each. Messenger-Edusoft, RD3 Box 462, Flemington, NJ 08822.

MICRO MEDIA MAGAZINE for Heath H-8, H-89 computers. Runs under HDOS or CP/M. Specify graphics or non-graphics, Microsoft or Benton Harbor Basic. Single bi-monthly issue is $11.95. Yearly rate is $55.00. M^3, 1316 Elmhurst, Garland, TX 75041.

STRUCTURED BASIC TRANSLATOR for Heath runs on Microsoft or B.H. Basic under HDOS. Adds compiler capabilities to interpreter basic. $29.95. Write M^3, 1316 Elmhurst, Garland, TX 75041.

QUALITY SOFTWARE. Database manager/report generator, advanced mailing list, inventory, word processor, GL, A/R, A/P and payroll systems. Available for TRS-80 Mod-I, II, CP/M and Heath. For catalog, send SASE (28¢) to Micro Architect Inc., 96 Dothan St., Arlington, MA 02174.

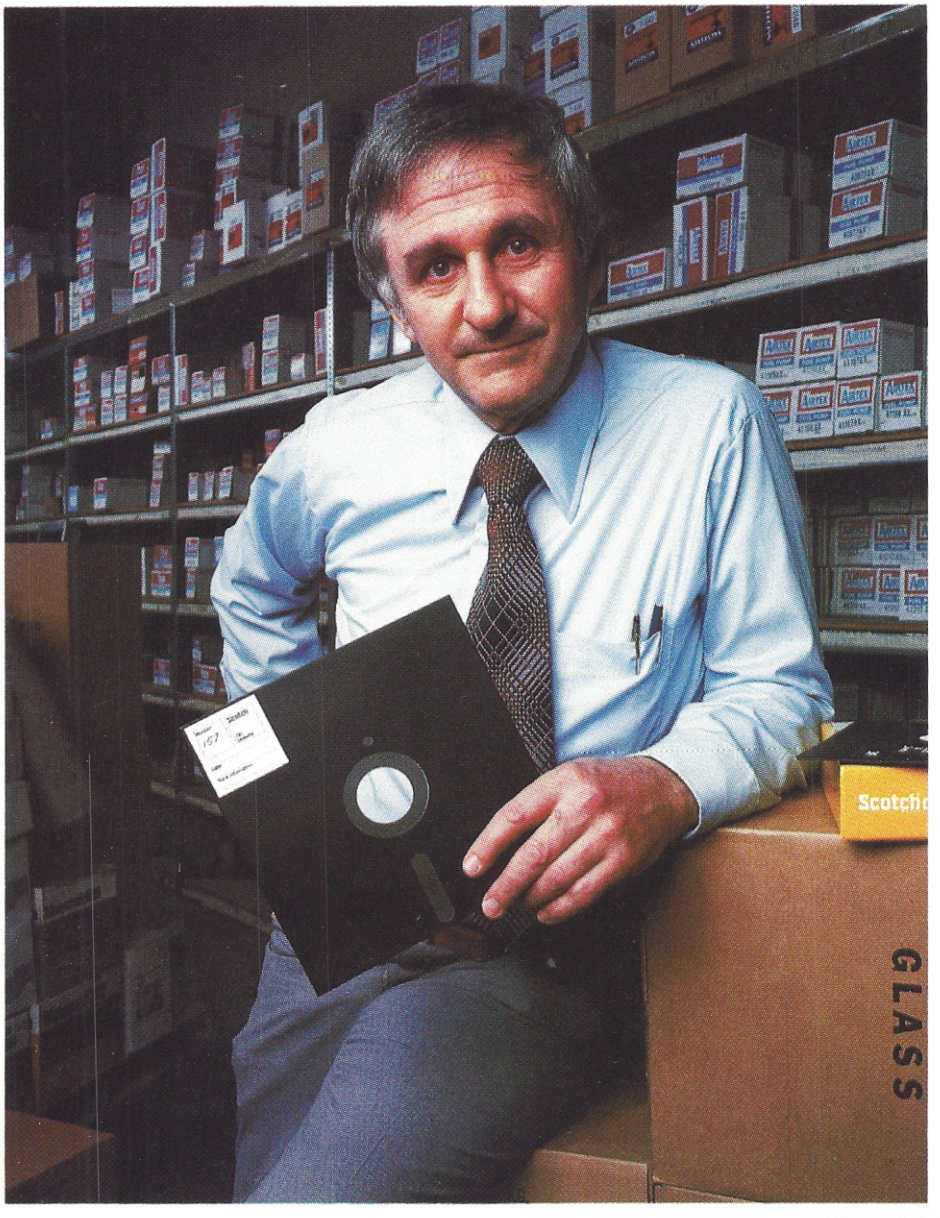
VIRTUOSO PET. 8/16/32K Commodores play Bach, Beethoven, etc. Music Box & Allen Animation have 4 programs, $10 each. Order both for free CB2 connector. Add $1 shipping. Allen Computer Products, Box 22839B, Livonia, MI 48151.

MICRODEX is a classified advertising section available to readers wishing to buy, sell or trade hardware and software. Price is $1 per word, with a 25 word minimum. The first two words bold. Additional bold type $5 per word.
Send copy with check or money order to INTERFACE AGE, Dept. M-I, 16704 Marquardt Ave., Cerritos, CA 90701.

ADVERTISER INDEX

*Manufacturer requests factory-direct inquiry.

"Our inventory is our existence. Think we'd trust it to anything less than Scotch® Brand Diskettes?"



Don Stone, President, Mass. Auto Supply Company, Inc., Boston, Mass.

Scotch Diskettes are the diskettes you can depend upon with the information *your* business depends upon.

Each one is tested and certified error-free before it leaves our factory. Because we know nothing less than perfection is acceptable for *your* vital business data.

Scotch Diskettes are available in regular or mini sizes, compatible with almost any system.

To find out where you can purchase Scotch Diskettes, call toll free: 800-328-1300. (In Minnesota, call collect: 612-736-9625.) Ask for the Data Recording Products Division. In Canada, write 3M Canada Inc., London, Ontario, N6A 4T1.

If it's worth remembering, it's worth Scotch Data Recording Products.

3M

Economy Sized Computers™ don't carry inflated price tags.

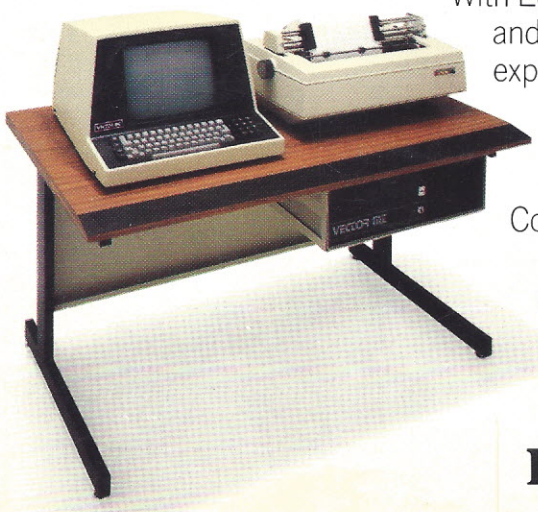
Think of this page as our competitor's price. And think of our ad as our price.

Ours is about ¼ the size of theirs. Pretty clever analogy, huh?

With Economy Sized Computers, you get data and word processing capabilities you only expect from much larger computers. All for the price of a new copying machine – just $10,463 suggested retail price.

So see the Vector Economy Sized Computer dealer near you.

And start improving the state of your economy.



CIRCLE INQUIRY NO. 69

VECTOR

VECTOR GRAPHIC INC.

Economy Sized Computers

31364 Via Colinas, Westlake Village, CA 91361, 213/991-